MAURÍCIO CORREIA LEMES NETO

Um Método dos Tablôs por Prova Direta para a Lógica Clássica

Florianópolis - SC

2004

UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO
EM CIÊNCIA DA COMPUTAÇÃO

Maurício Correia Lemes Neto

Um Método dos Tablôs por Prova Direta para a Lógica Clássica

Dissertação submetida à Universidade Federal de Santa Catarina como parte dos requisitos para a obtenção do grau de Mestre em Ciência da Computação.

Prof. Dr. Arthur Ronald de Vallauris Buchsbaum - Orientador

Florianópolis, agosto de 2004.

Um Método dos Tablôs por Prova Direta para a Lógica Clássica

Maurício Correia Lemes Neto

Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Ciência da Computação, área de concentração *Sistemas de Conhecimento*, e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Santa Catarina.

Prof. Raul Sidney Wazlawick, Dr.
Coordenador do Curso de Pós-Graduação em Ciência da Computação

Banca Examinadora

Prof. Arthur Ronald de Vallauris Buchsbaum, Dr.
Orientador (CPGCC - UFSC)

Prof. Décio Krause, Dr. - (Filosofia - UFSC)

Prof. Jorge Muniz Barreto, Dr. Sc. A. - (CPGCC - UFSC)

Prof. Paulo Sergio da Silva Borges, Dr. - (CPGCC - UFSC)

Inteligência é quando aprendemos com os próprios erros.
Sabedoria é quando aprendemos também com os erros dos outros.
(Anônimo)

Por que o caminho certo é mais difícil?
Talvez seja porque a recompensa no final é melhor.
(Anônimo)

Agradecimentos

Registro aqui o meu agradecimento primeiramente a Deus, por Ele sempre ser o meu fiel depositário, e também:

- Ao Curso de Pós-graduação em Ciência da Computação e à Universidade Federal de Santa Catarina, pela infra-estrutura e organização que viabilizaram o desenvolvimento deste trabalho;
- ao meu orientador, Prof. Dr. Arthur Ronald de Vallauris Buchsbaum, pelas valiosas idéias, dedicação, paciência e ensinamentos;
- aos professores do Curso de Pós-graduação em Ciência da Computação da UFSC, em especial aos professores Dr. Jorge Muniz Barreto (CPGCC-UFSC), Dr. Paulo Sergio da Silva Borges (CPGCC-UFSC), e Dr. Décio Krause (Filosofia - UFSC), que compuseram a banca examinadora;
- aos amigos que fiz no período que freqüentei as disciplinas do Curso, em especial aos membros do Laboratório de Conexionismo e Ciências Cognitivas;
- ao Centro Federal de Educação Tecnológica do Paraná, localizado na cidade de Cornélio Procópio, pois este foi o principal financiador deste trabalho;
- aos professores e amigos do Curso superior de Tecnologia em Informática do CEFET/PR, que foram os principais responsáveis pelo meu encaminhamento e incentivo;
- à minha família, que inclui minha querida e amada esposa (Eliane), pela dedicação e compreensão nos momentos tensos;
- aos meus pais (João e Eunice), os responsáveis diretos em me mostrar os caminhos corretos;
- às minhas irmãs (Maria Lucia, Sonia Ester, Sueli, Suzelena, Elizângela), que proporcionaram, na maioria das vezes em que estivemos juntos, alegrias e amor entre irmãos;
- a todos os meus queridos parentes.

Espero, de coração, que cada um possa gozar desta vitória e, juntamente comigo, se sentir, mais uma vez, um vencedor.

Sumário

Lista de Figuras

Tabela de Símbolos

Damos abaixo uma relação dos símbolos utilizados ao longo deste trabalho.

- $=$ – igualdade
- $\neq$ – diferença
- $\rightarrow$ – conectivo da implicação
- $\wedge$ – conectivo da conjunção
- $\vee$ – conectivo da disjunção
- $\neg$ – conectivo da negação
- $\leftrightarrow$ – conectivo da equivalência
- $\forall$ – quantificador universal
- $\exists$ – quantificador existencial
- $\overline{\exists}$ – quantificador, significa “existe um máximo”
- $\exists!$ – quantificador, significa “existe um único”
- $\surd$ – ticagem
- $\equiv$ – por definição
- $\vdash$ – barra de inferência, denota a relação de conseqüência para uma dada lógica
- $\subseteq$ – relação de inclusão
- $\supseteq$ – relação inversa da inclusão
- τ – artigo definido; $\tau \boldsymbol{x}\, \mathbf{P}$ significa, em geral, o objeto $\boldsymbol{x}$ tal que $\mathbf{P}$
- $\in$ – relação de pertinência
- $\mathbb{N}$ – conjunto dos números naturais
- $\mathbb{R}$ – conjunto dos números reais
- $\models_{\mathbf{LPC}}$ – relação de conseqüência especificada pela semântica da lógica proposicional clássica
- $\models_{\mathbf{LQC}}$ – relação de conseqüência especificada pela semântica da lógica quantificacional clássica

- min(**A**) – elemento mínimo de **A**
- max(**A**) – elemento máximo de **A**
- $\langle$**a**,**b**$\rangle$ – o par ordenado cujo primeiro componente é **a** e cujo segundo componente é **b**
- **A** $\times$ **B** – produto cartesiano de **A** por **B**
- $\mathcal{P}$(**A**) – conjunto potência de **A**
- D(**R**) – domínio de uma relação **R**
- $\mathcal{I}$(**R**) – imagem de uma relação **R**
- $\mathcal{C}$(**R**) – campo de uma relação **R**
- **R**$\langle$**A**$\rangle$ – imagem direta de **A** por **R**
- $\cup$ – sinal funcional para a união de conjuntos
- $\cap$ – sinal funcional para a intersecção de conjuntos
- ***f***: **A** $\rightarrow$ **B** – ***f*** é uma função de **A** em **B**
- ***f***: **A** $\rightarrowtail$ **B** – ***f*** é uma função parcial de **A** em **B**
- ***f***: **A** $\hookrightarrow$ **B** – ***f*** é uma transformação de **A** em **B**
- ***f***$\langle$**A**$\rangle$ – imagem direta de **A** por ***f***
- $\mathcal{C}(\rho)$ – função que atribui a ρ um dos valores “*fechado*” ou “*aberto*”
- **form**(η) – fórmula de um nó η
- **s**(***x***|**d**) – atribuição para variáveis que difere de **s** no máximo quanto à atribuição para ***x***
- σ(**c**|**d**) – mundo que difere de σ no máximo quanto à atribuição para a constante **c**
- **I**(***x***|**d**) – interpretação que difere de **I** no máximo quanto à atribuição para a constante **d**
- **I**(**c**|**d**) – interpretação que difere de **I** no máximo quanto à atribuição para a constante **d**

Tabela de Siglas

- **LPC** - Lógica Proposicional Clássica
- **LQC** - Lógica Quantificacional Clássica
- **STD** - Sistema de Tablôs por Prova Direta para **LPC**
- **STD*** - Sistema de Tablôs por Prova Direta para **LQC**

Tabela de Notações

Indicamos abaixo as convenções notacionais deste trabalho. No caso de letras individuais, valem os mesmos significados se as mesmas forem seguidas de subíndices ou plicas.

- $\mathbf{R}$ – relação; pg. 11
- $\boldsymbol{\mathsf{L}}$ – lógica arbitrária; pg. 21
- $\mathbf{L}$, $\mathbf{L}_1$, $\mathbf{L}_2$ – linguagens para $\boldsymbol{\mathsf{L}}$; pg. 21
- $\mathbf{P}$, $\mathbf{Q}$, $\mathbf{R}$, $\mathbf{S}$ – fórmulas; pg. 21
- Γ, Φ – coleções de fórmulas; pg. 21
- $\mathbf{v}$ – valor distinguido da semântica clássica, dito “verdadeiro”; pg. 23
- $\mathbf{f}$ – valor não distinguido da semântica clássica, dito “falso”; pg. 23
- Φ – alfabeto para a lógica proposicional clássica; pg. 24
- $\mathbf{L}_\Phi$ – linguagem proposicional; pg. 24
- Φ^* – alfabeto para a lógica quantificacional clássica; pg. 27
- $\mathbf{L}_{\Phi^*}$ – linguagem quantificacional; pg. 27
- $\boldsymbol{x}$, $\boldsymbol{y}$, $\boldsymbol{z}$, $\boldsymbol{w}$ – variáveis; pg. 27
- $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$ – constantes; pg. 27
- $\mathbf{f}$, $\mathbf{g}$, $\mathbf{h}$ – sinais funcionais; pg. 27
- $\mathbf{p}$, $\mathbf{q}$, $\mathbf{r}$ – sinais predicativos; pg. 27
- $\mathbf{t}$, $\mathbf{u}$, $\mathbf{v}$ – termos; pg. 27
- Δ – conjunto não vazio; pg. 30
- $\mathbf{d}$ – um elemento de Δ; pg. 30
- σ – mundo sobre Δ; pg. 30
- $\mathbf{L}$ e $\mathbf{L}$’ – linguagens formais; pg. 36
- $\mathbf{S}$ – sistema de tablôs; pg. 37
- $\mathbf{L}$ – linguagem inicial; pg. 37
- $\mathbf{L}$’ – linguagem de trabalho; pg. 37
- $\mathcal{I}$ – função de inicialização; pg. 37

- $\mathcal{C}$ – critério de fechamento; pg. 37
- $\mathcal{R}$ – conjunto de regras de expansão; pg. 37
- $\mathbf{T}$ – tablô; pg. 38
- $(\mathbf{T}_i)_{i \in I}$ – seqüência de desenvolvimento de tablôs; pg. 38
- "$\checkmark$" – sinal de checagem, significa que um nó já foi usado; pg. 43
- "$\smile$" – sinal que indica que um dado ramo está fechado; pg. 55
- "$\uparrow$" – sinal que indica que um dado ramo está aberto; pgs. 55 e 67
- $\mathbf{I}$ – interpretação; pg. 22
- σ – mundo; pg. 30

Convenções Gerais

Este trabalho é dividido em capítulos e cada capítulo é dividido em seções. Quando quisermos nos referir a teoremas, lemas, definições, corolários e outros enunciados e enumerados, dentro ou fora do capítulo em que estes ocorrerem, daremos o número do capítulo do enunciado citado seguido pelo seu número da seção. A contagem dos enunciados enumerados é reiniciada a cada seção. Por exemplo, o lema 3.2.2 refere-se ao segundo enunciado enumerado da segunda seção do capítulo 3, daí referir-nos-emos a este lema por lema 3.2.2.

As figuras são enumeradas de uma forma independente, e sua contagem é única dentro de cada capítulo, isto é, a sua contagem só e reinicializada no início de cada capítulo; assim, quando quisermos nos referir a uma figura, referir-nos-emos a ela dando o número do capítulo em que esta ocorre seguido pelo número da mesma. Por exemplo, a vigésima figura do capítulo 5 é citada por "Figura 5.20".

Indicamos o final de provas de teoremas possuindo um ou mais parágrafos próprios pela expressão "c.q.d.", a qual abrevia "conforme queria demonstrar".

RESUMO

Este trabalho desenvolve uma forma diferente de se obter árvores de prova por *tablôs*. Denominamos esse método de *direto*, por causa da característica em que a possível conclusão é inserida no *tablô inicial*, sem negá-la. Já o *método dos tablôs por refutação* se utiliza da negação da possível conclusão.

No sistema de tablôs por prova direta para a lógica clássica, cada ramo está relacionado semanticamente à disjunção das fórmulas que o compõem, e o tablô completo corresponde semanticamente à conjunção de todas essas disjunções. Em qualquer um dos métodos baseados em tablôs para a Lógica Clássica, tanto direto quanto indireto, um ramo é considerado fechado se o mesmo contiver duas fórmulas contraditórias. No método direto o fechamento de um ramo corresponde à sua *validade* semântica, a qual implica, no caso do fechamento de todos os ramos, na validade da possível conclusão. Já no método indireto o fechamento de um ramo corresponde à *insatisfatibilidade* da negação da possível conclusão, o que por sua vez implica na validade da mesma.

ABSTRACT

This work develops a different form of obtaining tree proofs by *tableaux*, which is denominated *direct*, due to the characteristic in which the possible conclusion is inserted in the *initial tableau*, without denying it. The *tableau method by refutation* uses instead the denial of the possible conclusion.

In the tableau system by direct proof for classical logic, each branch is related semantically to the disjunction of the formulas that compose it. The complete tableau corresponds semantically to the conjunction of all these disjunctions. For each one of the methods based on tableaux for classical logic, the direct or the indirect one, a branch is considered closed if the same contains two contradictory formulas. In the direct method the closure of a branch corresponds to its *validity*, which implies, in the case of the closing of all branches, the validity of the possible conclusion, whereas in the indirect method the closure of a branch corresponds to the *unsatisfiability* of the denial of the possible conclusion, which implies its validity.

Capítulo 1

Introdução

1.1 Breve Histórico

Em 1956 houve um evento organizado por um grupo de cientistas composto por Marvin Minsky, Herbert Simon, Allen Newell e John McCarthy, entre outros. Esse evento, intitulado “The Darthmouth Summer Research Project on Artificial Intelligence”, ocorreu no Dartmouth College, nos Estados Unidos.

Nessa conferência foi apresentado por Newell e Simon um sistema conhecido como LOGIC THEORIST, considerado na época um sistema revolucionário, pois conseguia provar certos teoremas da aritmética; com isso, tornou-se um dos primeiros programas empregados na manipulação de informações não numéricas.

Em 1950, A. M. Turing publicou um trabalho intitulado “Computing Machinery and Intelligence”, no qual perguntava-se acerca da capacidade de pensar das máquinas, pois, para muitos, o pensamento está diretamente ligado à inteligência, como na famosa frase de Descartes: “Penso, logo existo”. Passaram-se vários anos, mas a pergunta ainda continua.

Essa busca pelo conhecimento especializado proporciona uma constante evolução computacional, nos afetando de diversas formas, seja na vida pessoal, na vida profissional, nas tarefas executadas mecânica e diariamente, no modo de pensar, de agir e em ações de que nem nos damos conta que estamos executando.

Dentre essas buscas, existem tentativas de representar aspectos mecânicos do pensamento; acreditamos que estamos bem longe de alcançar tal proeza em sua plenitude. Pesquisadores ligados à Inteligência Artificial têm trabalhado com o objetivo de automatizar o que, para muitos seres humanos, é algo natural, tais como certas formas de raciocínio, dos quais podemos destacar o *dedutivo*. Segundo [15], um raciocínio dedutivo é aquele capaz de estabelecer a validade de um argumento deduzindo a sua conclusão a partir das suas premissas, mediante uma seqüência de raciocínios elementares.

A automatização do raciocínio dedica-se ao estudo de algoritmos capazes de inferir conclusões a partir de premissas previamente dadas. Estamos aqui especialmente interessados em alguma forma de dedução. Segundo [15], um argumento dedutivo é:

> “(...) aquele cujas premissas fornecem provas decisivas para a verdade de sua conclusão. Todo argumento dedutivo pode ser válido ou inválido: é válido na impossibilidade de suas premissas serem verdadeiras sem que também seja verdadeira a sua conclusão, e inválido no caso contrário. A teoria da dedução pretende explicar as relações entre as premissas e a conclusão de um raciocínio ou argumento válido e estabelecer técnicas para a avaliação dos argumentos dedutivos, isto é, para distinguir entre as deduções válidas e inválidas [pg. 139]”.

Formas de raciocínio capazes de explicitar um grande número de conclusões podem indicar algum grau de inteligência, e daí algoritmos capazes de implementá-las poderiam revelar-se bastante úteis na construção de sistemas computacionais capazes de simularem habilidades normalmente executadas por seres vivos dotados de formas de consciência razoavelmente evoluídas.

A lógica simbólica dedica-se ao estudo de sistemas formais, através de métodos matemáticos, que integram em uma unidade uma coleção de formas de raciocínio, aptas a interagir entre si de uma forma conjugada, sendo que a maioria de tais sistemas atende a diversas demandas de possíveis aplicações. Estamos especialmente interessados neste trabalho nos sistemas relacionados à Lógica Clássica, nos níveis proposicional e quantificacional, pois acreditamos que esta é uma base de acesso à compreensão de quase todos os sistemas lógicos mais importantes.

Entre os métodos de automatização do raciocínio podemos citar o *método da resolução* [34],[35], o *método dos seqüentes de Gentzen* [21], [44] e *o método dos tablôs* [38],[44].

O segundo envolve a manipulação formal de certos objetos sintáticos chamado de *seqüentes*, os quais compreendem uma lista de fórmulas denominadas *premissas*, seguidas de um sinal separador dito a barra de inferência ("$\vdash$"), e uma lista de fórmulas, ditas as *conclusões*.

O primeiro método, o da resolução, perfaz uma complexa manipulação formal inicial de fórmulas. O seu algoritmo correspondente principia com vários procedimentos destinados a reduzir as fórmulas dadas inicialmente a outras fórmulas em *forma prenex*[1], eliminando-se todos os quantificadores existenciais ("$\exists$") através de um método dito *skolemização*, bem como colocando na forma disjuntiva normal as subfórmulas livres de quantificadores.

Assim sendo, tanto o método da resolução como o método dos seqüentes não primam pela simplicidade, pois o primeiro envolve uma complexa manipulação simbólica antes do trabalho central do seu algoritmo correspondente, e o segundo não lida diretamente com fórmulas como objetos sintáticos básicos, mas sim com seqüentes.

[1]Uma fórmula prenex é uma fórmula da forma $\Psi_1\boldsymbol{x}_1,...,\Psi_n\boldsymbol{x}_n$ **P**, onde $\Psi_1,...,\Psi_n$ são um dos quantificadores "$\forall$" ou "$\exists$", e **P** não possui quantificadores.

Já o método dos tablôs destaca-se pela sua inerente simplicidade, pois, ao contrário do método dos seqüentes, lida diretamente com fórmulas, e, ao contrário do método da resolução, não exige complexos procedimentos iniciais de preparação das fórmulas envolvidas para que a parte central do seu algoritmo correspondente seja colocada em funcionamento.

1.2 Objetivo

Todas as apresentações que conhecemos dos métodos da resolução e dos tablôs envolvem uma análise semântica da negação da possível conclusão, sendo que ambos buscam demonstrar, nessa análise, a insatisfatibilidade dessa negação, mostrando que a suposição de sua satisfatibilidade conduz a um absurdo. Ambos os métodos envolvem, portanto, procedimentos de refutação da negação da possível conclusão.

Queremos mostrar que, no método dos tablôs, é possível proceder de uma outra forma, envolvendo uma análise semântica direta da conclusão, sem qualquer refutação, mas sim através de uma justificativa direta da validade da conclusão.

1.3 Discussão e Justificativa do Tema

Os três métodos de automatização citados são computacionalmente implementados, sendo que tanto o método dos tablôs quanto o dos seqüentes utilizam-se de certas estruturas de dados chamadas de árvores, com a diferença de que, no método dos tablôs, os nós são constituídos por fórmulas, enquanto que no método dos seqüentes os nós são constituídos por seqüentes.

Escolhemos o método dos tablôs em detrimento dos métodos dos seqüentes de Gentzen e da resolução como forma de automatização do raciocínio pelas duas seguintes razões:

(1) Os seus objetos básicos de manipulação são fórmulas, enquanto que o método dos seqüentes de Gentzen utiliza-se de seqüentes, que são em si objetos bem mais complexos que fórmulas.

(2) Após um procedimento de inicialização bem simples, o método dos tablôs conduz diretamente ao seu algoritmo principal, enquanto que, no método da resolução, o procedimento de inicialização é muito mais complexo.

Os métodos dos tablôs e da resolução são, em sua forma tradicional, baseados na refutação da negação da possível conclusão, enquanto que o método dos seqüentes de Gentzen é direto, pois lida diretamente com o possível seqüente válido. Isto poderia indicar uma possível superioridade do método dos seqüentes de Gentzen sobre o método dos tablôs, o que é contestado nesta dissertação, na qual mostramos que o método dos tablôs pode ser trabalhado de uma forma direta pelo menos tão boa quanto o método dos tablôs por refutação.

Em muitas ocasiões, quando uma dada asserção é repetida exaustivamente dentro de uma dada comunidade esta pode passar a reputá-la como verdadeira. Isto é o que vem acontecendo com os métodos dos tablôs e da resolução, sobre os quais todas as referências por nós consultadas classificam ambos como sendo métodos de prova por refutação.

Este trabalho busca superar tal preconceito.

1.4 Estrutura do Trabalho

Este trabalho está organizado em capítulos, cujos conteúdos são abordados na tentativa de conduzir de forma clara e concisa diversos assuntos necessários e básicos para o entendimento da proposta principal aqui exposta.

Descreveremos de forma sucinta os capítulos e os assuntos tratados em cada um deles.

O segundo capítulo, **Noções Básicas de Teoria dos Conjuntos**, fornece de forma sucinta os pré-requisitos matemáticos gerais para o desenvolvimento deste trabalho, incluindo a definição de árvore e das principais idéias a ela associadas.

O terceiro capítulo, **A Lógica Clássica**, expõe brevemente as principais idéias das semânticas de valorações, e define a semântica clássica nos níveis proposicional e quantificacional, bem como alguns lemas elementares, indispensáveis para o desenvolvimento posterior deste trabalho.

O quarto capítulo, **O Método dos Tablôs**, desenvolve, de forma generalizada, os principais tópicos que dizem respeito ao método dos tablôs, estabelecendo em seguida a exposição das condições gerais de correção e completude para uma lógica qualquer dotada de uma semântica de valorações, segundo o paradigma do método de prova direta.

O quinto capítulo, **Um Sistema de Tablôs por Prova Direta para a Lógica Proposicional Clássica**, define um sistema de tablôs por prova direta para a lógica proposicional clássica, e demonstra a validade das condições gerais da correção e completude para este sistema. A título de ilustração, são dados dois exemplos.

O sexto capítulo, **Um Sistema de Tablôs por Prova Direta para a Lógica Quantificacional Clássica**, é uma extensão para a lógica quantificacional do sistema exposto no capítulo anterior. Também é demonstrado aqui a adequação das condições gerais de correção e completude para este sistema. No final são dados dois exemplos a título de ilustração.

Finalmente, o último capítulo, **Considerações Finais**, dá as nossas conclusões pessoais, bem como sugere algumas linhas futuras de pesquisa.

Capítulo 2

Noções Básicas de Teoria dos Conjuntos

2.1 Introdução

Uma *árvore* é uma coleção não vazia de elementos ditos *nós*, munida de uma *relação de ordem estrita*, e de uma função que, aplicada a cada nó, fornece o seu *conteúdo*. É dotada de um elemento mínimo, dito o *nó raiz* desta árvore, tal que cada nó distinto do nó raiz possui um único predecessor.

Uma estrutura de dados essencial para a definição de *tablô* é a estrutura de árvore. Tal estrutura requer, por sua vez, para ser bem definida, as idéias de relação e de função, entre outras. Faremos daí uma breve exposição de algumas idéias da teoria dos conjuntos que consideramos fundamentais para este trabalho. Exposições mais detalhadas deste assunto podem ser vistas em [22] e [27].

2.1.1 Definição: Um *par ordenado* é uma lista ordenada de dois elementos **a** e **b**. Notamos um par ordenado de dois elementos **a** e **b** por $\langle \mathbf{a},\mathbf{b} \rangle$, onde **a** é dito a *primeira coordenada* ou *abscissa*, e **b** é dito a *segunda coordenada* ou *ordenada do par*.

Podemos reduzir a idéia de par ordenado à idéia de conjunto, ou podemos considerar par ordenado como um conceito primitivo de uma dada teoria dos conjuntos. É essencial, no entanto, que, dados dois pares ordenados $\langle$**a**,**b**$\rangle$ e $\langle$**a'**,**b'**$\rangle$, a seguinte proposição seja válida.

2.1.2 Proposição: (Igualdade de Pares Ordenados)
• $\vdash$ $\langle$**a**,**b**$\rangle$ = $\langle$**a'**,**b'**$\rangle$ $\leftrightarrow$ **a** = **a'** $\wedge$ **b** = **b'**.

Definimos de uma forma análoga uma lista ordenada de **n** elementos $\mathbf{a}_1,\ldots,\mathbf{a}_n$, e notamos tal lista por $\langle \mathbf{a}_1,\ldots,\mathbf{a}_n \rangle$.

2.1.3 Proposição: (Igualdade de Listas Ordenadas)
• $\vdash \langle \mathbf{a}_1,\ldots,\mathbf{a}_n \rangle = \langle \mathbf{b}_1,\ldots,\mathbf{b}_n \rangle \leftrightarrow \mathbf{a}_1 = \mathbf{b}_1 \wedge \cdots \wedge \mathbf{a}_n = \mathbf{b}_n$.

2.1.4 Definição: (Produto Cartesiano)
Dados dois conjuntos **A** e **B**, o *produto cartesiano de* **A** *e* **B**, notado por $\mathbf{A} \times \mathbf{B}$, é definido por $\mathbf{A} \times \mathbf{B} \equiv \{\langle\, x, y \,\rangle \mid x \in \mathbf{A} \wedge y \in \mathbf{B}\,\}$.

2.1.5 Definição: (Conjunto Potência)
Dado um conjunto **A**, o *conjunto potência de* **A**, notado por $\mathcal{P}(\mathbf{A})$, é o conjunto de todos os subconjuntos de **A**.

2.1.6 Definição: (Relação)
Um conjunto é dito uma *relação* se todos os seus elementos forem pares ordenados.

2.1.7 Definição: **R** é dito uma *relação em* **A** se $\mathbf{R} \subseteq \mathbf{A} \times \mathbf{A}$.

2.1.8 Definição: **R** é dito uma *relação de* **A** *em* **B** se $\mathbf{R} \subseteq \mathbf{A} \times \mathbf{B}$.

2.1.9 Definição: Dizemos que x *está relacionado com* y *em* **R**, e notamos isto por $x\ \mathbf{R}\ y$, se $\langle x, y \rangle \in \mathbf{R}$.

2.1.10 Definição: Seja **R** uma relação. Dizemos que x *não está relacionado com* y *em* **R**, e notamos isto por $x \not\mathbf{R}\ y$, se $\neg(x\ \mathbf{R}\ y)$.

Para facilitar comentários posteriores que eventualmente faremos acerca da pertinência de pares ordenados em uma relação, damos a seguir duas novas terminologias.

2.1.11 Definição: Sejam **R** uma relação, e considere que $\boldsymbol{x}\ \mathbf{R}\ \boldsymbol{y}$.

- Dizemos neste caso que $\boldsymbol{x}$ *minora* $\boldsymbol{y}$ *em* **R**, ou que $\boldsymbol{x}$ ***R**-minora* $\boldsymbol{y}$, ou que $\boldsymbol{x}$ é um *minorante de* $\boldsymbol{y}$ *em* **R**, ou também que $\boldsymbol{x}$ é um ***R**-minorante de* $\boldsymbol{y}$.
- Dizemos também que $\boldsymbol{y}$ *majora* $\boldsymbol{x}$ *em* **R**, ou que $\boldsymbol{y}$ **R**-*majora* $\boldsymbol{x}$, ou que $\boldsymbol{y}$ é um *majorante de* $\boldsymbol{x}$ *em* **R**, ou também que $\boldsymbol{y}$ é um **R**-*majorante de* $\boldsymbol{x}$.

2.1.12 Definição: (Domínio de uma Relação)
Se **R** é uma relação, *o domínio de* **R** é a coleção de todos os elementos que minoram algum objeto em **R**, isto é, $\mathrm{D}(\mathbf{R}) \equiv \{\boldsymbol{x} \mid \exists \boldsymbol{y}\ (\boldsymbol{x}\ \mathbf{R}\ \boldsymbol{y})\}$

2.1.13 Definição: (Imagem de uma Relação)
Se **R** é uma relação, *a imagem de* **R** é a coleção de todos os elementos que majoram algum objeto em **R**, isto é, $\mathcal{I}(\mathbf{R}) \equiv \{\boldsymbol{y} \mid \exists \boldsymbol{x}\ (\boldsymbol{x}\ \mathbf{R}\ \boldsymbol{y})\}$.

2.1.14 Definição: (Campo de uma Relação)
O *campo de* **R** é a coleção de todos os elementos que minoram ou que majoram algum objeto em **R**, isto é, $\mathcal{C}(\mathbf{R}) \equiv \{\boldsymbol{z} \mid \exists \boldsymbol{x}\ (\boldsymbol{x}\ \mathbf{R}\ \boldsymbol{z}\ ) \vee \exists \boldsymbol{y}\ (\ \boldsymbol{z}\ \mathbf{R}\ \boldsymbol{y})\}$.

2.1.15 Definição: (Imagem Direta de um Conjunto por uma Relação)
Sejam **R** uma relação e **A** um conjunto. A *imagem direta de* **A** *por* **R**, notada por $\mathbf{R}\langle\mathbf{A}\rangle$, é a coleção de todos os objetos que majoram algum elemento de **A** em **R**, isto é, $\mathbf{R}\langle\mathbf{A}\rangle \equiv \{\boldsymbol{y} \mid \exists \boldsymbol{x}\ (\boldsymbol{x} \in \mathbf{A} \wedge \boldsymbol{x}\ \mathbf{R}\ \boldsymbol{y})\}$

2.1.16 Teorema: $\vdash \mathcal{C}(\mathbf{R}) = \mathrm{D}(\mathbf{R}) \cup \mathcal{I}(\mathbf{R})$.

2.1.17 Definição: (Tipos de Relação com respeito a um Conjunto Dado)
Sejam **R** uma relação e **A** um conjunto.

- **R** é reflexiva em $\mathbf{A} \equiv \forall \boldsymbol{x}(\boldsymbol{x} \in \mathbf{A} \rightarrow \boldsymbol{x}\ \mathbf{R}\ \boldsymbol{x})$.
- **R** é irreflexiva em $\mathbf{A} \equiv \forall \boldsymbol{x}(\boldsymbol{x} \in \mathbf{A} \rightarrow \boldsymbol{x}\ \not\mathbf{R}\ \boldsymbol{x})$.
- **R** é simétrica em $\mathbf{A} \equiv \forall \boldsymbol{x}\ \forall \boldsymbol{y}(\boldsymbol{x}, \boldsymbol{y} \in \mathbf{A} \wedge \boldsymbol{x}\ \mathbf{R}\ \boldsymbol{y} \rightarrow \boldsymbol{y}\ \mathbf{R}\ \boldsymbol{x})$.
- **R** é assimétrica em $\mathbf{A} \equiv \forall \boldsymbol{x}\ \forall \boldsymbol{y}(\boldsymbol{x}, \boldsymbol{y} \in \mathbf{A} \wedge \boldsymbol{x}\ \mathbf{R}\ \boldsymbol{y} \rightarrow \boldsymbol{y}\ \not\mathbf{R}\ \boldsymbol{x})$.
- **R** é anti-simétrica em $\mathbf{A} \equiv \forall \boldsymbol{x}\ \forall \boldsymbol{y}(\boldsymbol{x}, \boldsymbol{y} \in \mathbf{A} \wedge \boldsymbol{x}\ \mathbf{R}\ \boldsymbol{y} \wedge \boldsymbol{y}\ \mathbf{R}\ \boldsymbol{x} \rightarrow \boldsymbol{x} = \boldsymbol{y})$.
- **R** é transitiva em $\mathbf{A} \equiv \forall \boldsymbol{x}\ \forall \boldsymbol{y}\ \forall \boldsymbol{z}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z} \in \mathbf{A} \wedge \boldsymbol{x}\ \mathbf{R}\ \boldsymbol{y} \wedge \boldsymbol{y}\ \mathbf{R}\ \boldsymbol{z} \rightarrow \boldsymbol{x}\ \mathbf{R}\ \boldsymbol{z})$.
- **R** é uma relação de equivalência em $\mathbf{A} \equiv$ **R** é uma relação reflexiva, simétrica e transitiva em **A**.

Os exemplos expostos a seguir são referentes a cada conceito exposto 2.1.17. Estes foram parcialmente baseados em [43].

2.1.18 Exemplo de Relação Reflexiva

Sejam **A** uma coleção de conjuntos e **R** uma relação em **A** definida por "x é um subconjunto de y". Como todo conjunto é um subconjunto de si próprio, temos que **R** é uma relação reflexiva em **A**.

2.1.19 Exemplo de Relação Simétrica

Sejam **A** um conjunto de seres humanos e **R** uma relação simétrica em **A** definida por "x é parente de y". Se x, y são seres humanos e x é parente de y, então, obviamente, y é parente de x.

2.1.20 Exemplo de Relação Não Simétrica

Seja **R** a relação em $\mathbb{N}$ definida por "x divide y" (o que notamos por x/y). Temos que **R** não é simétrica em **A**, pois 4 e 8 são números naturais tal que $4/8$, mas $\neg(8/4)$.

2.1.21 Exemplo de Relação Anti-simétrica

Seja **R** a relação em $\mathbb{R}$ definida por "$x \leq y$". Como, para quaisquer números reais x e y, $x \leq y$ e $y \leq x$ implica que $x = y$, temos que **R** é uma relação anti-simétrica em $\mathbb{R}$.

2.1.22 Exemplo de Relação Transitiva

Seja **R** a relação em $\mathbb{R}$ definida por "x é menor que y". Temos que **R** é transitiva, pois, dados números reais x, y, z, se $x < y$ e $y < z$, então $x < z$.

2.1.23 Exemplo de Relação de Equivalência

Sejam **A** um conjunto qualquer, e **R** uma relação em **A** definida por "x é igual a y". Temos que **R** é uma relação de equivalência em **A**, pois **R** é reflexiva, simétrica e transitiva em **A**.

2.1.24 Definição: (Relação de Ordem)

Dizemos que **R** é uma *relação de ordem em* **A**, ou simplesmente que **R** é uma *ordem em* **A**, se **R** é uma relação anti-simétrica e transitiva em **A**.

2.1.25 Definição: (Relação de Ordem Estrita)

Dizemos que **R** é uma *relação de ordem estrita em* **A**, ou simplesmente que **R** é uma *ordem estrita em* **A**, se **R** é uma ordem em **A** e **R** é irreflexiva em **A**.

2.1.26 Definição: (Relação Linear)

- Dizemos que **R** é uma *relação linear em* **A** se $\forall x\ \forall y\ (x,y \in \mathbf{A} \wedge x \neq y \rightarrow x\ \mathbf{R}\ y \vee y\ \mathbf{R}\ x)$.

Certos elementos de um dado conjunto munido de uma relação de ordem possuem algumas características notáveis, que serão dadas a seguir.

2.1.27 Notação: No restante desta seção, **R** é uma relação de ordem em **A**.

O *elemento* **R**-*mínimo em* **A** é o elemento de **A** que **R**-minora todos os outros elementos de **A**.

2.1.28 Definição: (R-mínimo em A)

Seja $x \in \mathbf{A}$.
Dizemos que x é o **R**-*mínimo em* **A** se $\forall y(y \in \mathbf{A} \wedge x \neq y \rightarrow x\ \mathbf{R}\ y)$.

2.1.29 Definição: (Elemento R-mínimo de A)

Se **A** é um conjunto dotado de um elemento **R**-mínimo, notamos este por $\text{mínimo}_{\mathbf{R}}(\mathbf{A})$. Caso a relação **R** esteja implícita, notamos $\text{mínimo}_{\mathbf{R}}(\mathbf{A})$ apenas por mínimo(**A**).

O *elemento* **R**-*máximo em* **A** é o elemento de **A** que **R**-majora todos os outros elementos de **A**.

2.1.30 Definição: (R-máximo em A)

Seja $x \in \mathbf{A}$.
Dizemos que x é o **R**-*máximo em* **A** se $\forall y(y \in \mathbf{A} \wedge x \neq y \rightarrow y\ \mathbf{R}\ x)$.

2.1.31 Definição: (Elemento R-máximo de A)

Se **A** é um conjunto de um elemento **R**-máximo, notamos este por $\text{máximo}_{\mathbf{R}}(\mathbf{A})$. Caso a relação **R** esteja implícita, notamos $\text{máximo}_{\mathbf{R}}(\mathbf{A})$ apenas por máximo(**A**).

Um *elemento* **R**-*maximal em* **A** é um elemento de **A** que não é *majorado* em **R** por nenhum outro elemento de **A**.

2.1.32 Definição: (Elemento R-maximal em A)
Seja $\boldsymbol{y} \in \mathbf{A}$.
$\boldsymbol{y}$ é dito *um elemento* **R**-*maximal em* **A** se $\forall \boldsymbol{x}\ (\ \boldsymbol{x} \in \mathbf{A} \wedge \boldsymbol{x} \neq \boldsymbol{y} \rightarrow \neg(\boldsymbol{y}\ \mathbf{R}\ \boldsymbol{x}))$.

Um *elemento* **R**-*minimal em* **A** é um elemento de **A** que não é *minorado* em **R** por nenhum outro elemento de **A**.

2.1.33 Definição: (Elemento R-minimal em A)
Seja $\boldsymbol{y} \in \mathbf{A}$.
$\boldsymbol{y}$ é dito *um elemento* **R**-*minimal em* **A** se $\forall \boldsymbol{x}\ (\ \boldsymbol{x} \in \mathbf{A} \wedge \boldsymbol{x} \neq \boldsymbol{y} \rightarrow \neg(\boldsymbol{x}\ \mathbf{R}\ \boldsymbol{y}))$.

Um **R**-*predecessor em* **A** *de um dado* $\boldsymbol{x} \in \mathbf{A}$ é um **R**-minorante de $\boldsymbol{x}$ distinto de $\boldsymbol{x}$, **R**-maximal com respeito à coleção dos **R**-minorantes de $\boldsymbol{x}$ em **A** distintos de $\boldsymbol{x}$.

2.1.34 Definição: (R-predecessor de um dado elemento em A)
Sejam $\boldsymbol{x}, \boldsymbol{y} \in \mathbf{A}$. $\boldsymbol{x}$ é dito um **R**-*predecessor de* $\boldsymbol{y}$ *em* **A** se $\boldsymbol{x}\ \mathbf{R}\ \boldsymbol{y}$, $\boldsymbol{x} \neq \boldsymbol{y}$ e $\forall \boldsymbol{x}'(\boldsymbol{x}' \in \mathbf{A} \wedge \boldsymbol{x}'\ \mathbf{R}\ \boldsymbol{y} \rightarrow \neg(\boldsymbol{x}\ \mathbf{R}\ \boldsymbol{x}'))$.

Um **R**-*sucessor em* **A** *de um dado* $\boldsymbol{x} \in \mathbf{A}$ é um **R**-majorante de $\boldsymbol{x}$ distinto de $\boldsymbol{x}$, **R**-minimal com respeito à coleção dos **R**-majorantes de $\boldsymbol{x}$ em **A** distintos de $\boldsymbol{x}$.

2.1.35 Definição: (R-sucessor de um dado elemento em A)
Sejam $\boldsymbol{x}, \boldsymbol{y} \in \mathbf{A}$. $\boldsymbol{y}$ é dito um **R**-*sucessor de* $\boldsymbol{x}$ *em* **A** se $\boldsymbol{x}\ \mathbf{R}\ \boldsymbol{y}$, $\boldsymbol{x} \neq \boldsymbol{y}$ e $\forall \boldsymbol{y}'(\boldsymbol{y}' \in \mathbf{A} \wedge \boldsymbol{x}\ \mathbf{R}\ \boldsymbol{y}' \rightarrow \neg(\boldsymbol{y}'\ \mathbf{R}\ \boldsymbol{y}))$.

2.1.36 Definição: (Função)
Uma relação $\boldsymbol{f}$ é dita uma *função* se $\forall \boldsymbol{x}\ \overline{\exists} \boldsymbol{y}(\boldsymbol{x}\ \boldsymbol{f}\ \boldsymbol{y})$.

2.1.37 Escólio: Se $\boldsymbol{f}$ é uma função, então as seguintes proposições são verdadeiras:
- $\forall \boldsymbol{x}\ \forall \boldsymbol{y}\ \forall \boldsymbol{y}'\ (\boldsymbol{x}\ \boldsymbol{f}\ \boldsymbol{y} \wedge \boldsymbol{x}\ \boldsymbol{f}\ \boldsymbol{y}' \rightarrow \boldsymbol{y} = \boldsymbol{y}')$.
- $\forall \boldsymbol{x} \in \mathrm{D}(\boldsymbol{f})\ \exists! \boldsymbol{y}(\boldsymbol{x}\ \boldsymbol{f}\ \boldsymbol{y})$.

2.1.38 Definição: (Aplicação de uma Função a um Elemento x)
Se $\boldsymbol{f}$ é uma função e $\boldsymbol{x} \in \mathrm{D}(\boldsymbol{f})$, definimos *a aplicação de* $\boldsymbol{f}$ *a* $\boldsymbol{x}$, e notamos isto por $\boldsymbol{f}(\boldsymbol{x})$, por $\boldsymbol{f}(\boldsymbol{x}) \equiv \tau \boldsymbol{y}\ (\boldsymbol{x}\ \boldsymbol{f}\ \boldsymbol{y})$.

2.1.39 Definição: (Função de A em B)
$\boldsymbol{f}$ é dito uma *função de* $\mathbf{A}$ *em* $\mathbf{B}$, e notamos isto por $\boldsymbol{f}\colon \mathbf{A} \to \mathbf{B}$, se $\boldsymbol{f}$ é uma função, $\mathrm{D}(\boldsymbol{f}) = \mathbf{A}$ e $\mathcal{I}(\boldsymbol{f}) \subseteq \mathbf{B}$.

2.1.40 Definição: (Função Parcial de A em B)
$\boldsymbol{f}$ é dito uma *função parcial de* $\mathbf{A}$ *em* $\mathbf{B}$, e notamos isto por $\boldsymbol{f}\colon \mathbf{A} \rightarrowtail \mathbf{B}$, se $\boldsymbol{f}$ é uma função, $\mathbf{A} \supseteq \mathrm{D}(\boldsymbol{f})$ e $\mathcal{I}(\boldsymbol{f}) \subseteq \mathbf{B}$.

2.1.41 Definição: (Transformação de A em B)
$\boldsymbol{f}$ é dito uma *transformação de* $\mathbf{A}$ *em* $\mathbf{B}$, e notamos isto por $\boldsymbol{f}\colon \mathbf{A} \hookrightarrow \mathbf{B}$, se $\boldsymbol{f}$ é uma função, $\mathrm{D}(\boldsymbol{f}) \supseteq \mathbf{A}$ e $\boldsymbol{f}\langle\mathbf{A}\rangle \subseteq \mathbf{B}$.

2.2 Árvores

Uma árvore é uma coleção não vazia de elementos denominados nós, munida de uma relação de ordem estrita e de uma função que, aplicada a cada nó, dá o seu conteúdo, na qual há um elemento mínimo, chamado de raiz, e todo nó distinto da raiz possui um único predecessor.

2.2.1 Definição: Uma *árvore* é um terno $\mathbf{T} = \langle \mathbf{N},\mathbf{A},\mathbf{C} \rangle$ obedecendo às seguintes condições:

- $\mathbf{N}$ é um conjunto não vazio, cujos elementos são ditos *nós em* $\mathbf{T}$.
- $\mathbf{A}$ é uma relação de ordem estrita em $\mathbf{N}$, cujo campo contém $\mathbf{N}$, dita *a relação de ascendência em* $\mathbf{T}$; se η,η' são dois nós em $\mathbf{T}$ e $\eta\ \mathbf{A}\ \eta$', dizemos que η é *um ascendente de* η' *em* $\mathbf{T}$, ou que η' é um *descendente de* η *em* $\mathbf{T}$[1].
- Existe um único nó em $\mathbf{T}$ que não possui ascendentes em $\mathbf{T}$, e que, por sua vez, é ascendente de todos os outros nós em $\mathbf{T}$, dito *o nó raiz*, ou simplesmente *a raiz em* $\mathbf{T}$.
- Todo nó em $\mathbf{T}$ distinto da raiz possui um único $\mathbf{A}$-predecessor em $\mathbf{N}$, dito *o seu pai em* $\mathbf{T}$.
- $\mathbf{C}$ é uma função cujo domínio contém $\mathbf{N}$ tal que, para cada nó η em $\mathbf{T}$, $\mathbf{C}(\eta)$ é dito *o conteúdo de* η *em* $\mathbf{T}$.

[1]Na linguagem geral de relações, η minora η' em $\mathbf{A}$, e η' majora η em $\mathbf{A}$.

2.2.2 Definição: Uma *árvore* é dita *finita* se a sua coleção de nós for um conjunto finito.

2.2.3 Teorema: Se **T** = $\langle$**N**,**A**,**C**$\rangle$ é uma árvore, então as seguintes proposições são equivalentes:
- η é a raiz em **T**;
- η é o **A**-mínimo em **N**.

Podemos representar graficamente uma árvore de diferentes formas, dentre elas destacamos: conjuntos aninhados, barras ou tabelas, parênteses aninhados ou certos tipos particulares de grafos. Neste trabalho será usada a representação gráfica apresentada na figura 2.1.

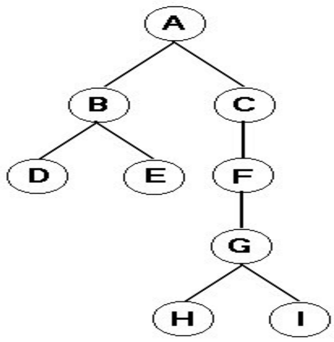
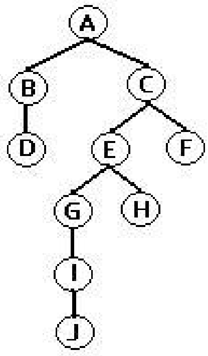


Figura 2.1: Árvore

Tal representação facilita a compreensão de como elas são ordenadas; a sua organização reflete uma situação oriunda de uma extensão por sucessões. Nas figuras 2.1 e 2.2 o menor elemento é aquele que está no topo, denominado raiz da árvore. Dado um nó de uma destas figuras, e seguindo as arestas do grafo, podemos visitar qualquer outro nó do qual este é ascendente caminhando para cima a partir do nó dado; de uma forma análoga, podemos acessar um nó descendente deste nó caminhando para baixo. Abusando do vocabulário e da percepção do leitor, as árvores aqui expostas são vistas de “cabeça para baixo”; isso significa que o nó raiz está no topo e o crescimento é dado no sentido vertical para baixo.



Nó	Aridade
A	2
B	1
C	2
D	0
E	2
F	0
G	1
H	0
I	1
J	0

Figura 2.2: Aridades de nós de uma árvore

2.2.4 Definição: (Filhos e Folhas)

Seja $\mathbf{T} = \langle \mathbf{N}, \mathbf{A}, \mathbf{C} \rangle$ uma árvore, e η,η' nós de **T**.

- Dizemos que η é *um filho de* η*' em* **T** se η é um **A**-sucessor de η' em **N**.
- Dizemos que η é *uma folha em* **T** se η é um elemento **A**-maximal em **N**.

2.2.5 Definição: (Nível de um nó em uma árvore)

Seja $\mathbf{T} = \langle \mathbf{N}, \mathbf{A}, \mathbf{C} \rangle$ uma árvore.

Especificamos por recursão uma função $\mathcal{N}$ de $\mathbb{N}$ em $\mathcal{P}(\mathbf{N})$[2] pelas seguintes cláusulas:

- $\mathcal{N}(0)$ é um conjunto unitário cujo único elemento é o nó raiz em **T**.
- $\mathcal{N}(\mathbf{k}+1)$ é a coleção de todos os filhos em **T** de algum elemento de $\mathcal{N}(\mathbf{k})$.
- Se $\mathbf{k} \in \mathbb{N}$, dizemos que um nó η em **T** é de nível **k** em **T** se $\eta \in \mathcal{N}(\mathbf{k})$.

2.2.6 Definição: (Profundidade de uma árvore)

Dizemos que **p** é a *profundidade de uma árvore* se **p** é o maior número natural tal que $\mathcal{N}(\mathbf{p})$ é um conjunto não vazio.

2.2.7 Definição: (Aridade de um nó em uma árvore)

Dizemos que o número natural **k** é a *aridade de um dado nó* em uma árvore se este nó tiver exatamente **k** filhos nesta árvore.

2.2.8 Definição: (Árvore de ramificação finita)

Dizemos que **T** é uma árvore *de ramificação finita* se cada um de seus nós que não for folha possuir um número finito de filhos.

2.2.9 Escólio: Existem árvores cuja profundidade não pode ser mensurada por um número natural.

[2]Conjunto Potência de **N**.

2.2.10 Definição: (Aridade de uma árvore)
Seja **T** uma árvore. Dizemos que o número natural **k** é a *aridade de* **T** se **k** é o maior número natural tal que existe um nó em **T** cuja aridade é **k**[3].

2.2.11 Escólio: Existem árvores que, embora sendo de ramificação finita, não possuem aridade.

2.2.12 Definição: (Ramo de uma árvore)
Seja uma árvore **T** = $\langle$**N**,**A**, **C**$\rangle$ e $\mathcal{L} = \{\rho \in \mathcal{P}(\mathbf{N}) \mid \mathbf{A}$ é uma relação de ordem linear em $\rho\}$. Um elemento $\subseteq$-maximal em $\mathcal{L}$ é dito um ramo em **T**.

2.2.13 Escólio: Um ramo de uma árvore é *linearmente ordenado* pela sua relação de ascendência[4].

2.2.14 Definição: Um ramo de uma árvore é dito *finito* se este for uma coleção finita.

2.2.15 Definição: (k-segmento)
Seja **k** um número natural.
Chamamos a coleção de todos os números naturais menores ou iguais a **k** de ***k**-segmento*.

2.2.16 Teorema: Seja **T** = $\langle$**N**,**A**, **C**$\rangle$ uma árvore. As seguintes proposições são equivalentes:
- ρ é um ramo em **T**.
- Existe uma função $\boldsymbol{f}$ cujo domínio é $\mathbb{N}$ ou um **k**-segmento, para algum número natural **k** fixo tal que $\rho = \mathcal{I}(\boldsymbol{f})$ e, para todo $\mathbf{k} \in D(\boldsymbol{f})$ tal que $\mathbf{k}+1 \in D(\boldsymbol{f})$, $\boldsymbol{f}(\mathbf{k}+1)$ é um filho de $\boldsymbol{f}(\mathbf{k})$ em **T**.

[3]Em outras palavras, este nó possui exatamente **k** filhos.

[4]Isto é, a relação de ascendência é uma relação de ordem linear no ramo.

Capítulo 3

A Lógica Clássica

3.1 Introdução

> "...A lógica, como ciência, estuda as manifestações da razão operativa em todos os contextos lingüísticos possíveis, nas linguagens escrita e falada. O estudo de tais manifestações, tais como ocorrem na vida quotidiana, sem nenhuma idealização, é feito por um ramo desta ciência dito *Lógica Informal*. Um estudo idealizado de tais manifestações, lançando mão de *linguagens formais* e métodos matemáticos, é feito pelo ramo dito *Lógica Formal*..."[13].

O raciocínio ou inferência é um processo que parte de um conhecimento explícito, dado pelas premissas, para um conhecimento em geral implícito, dado pelas conclusões.

Atualmente existem vários sistemas lógicos concebidos para representar as diversas formas de raciocínio. No século IV A.C., Aristóteles, um filósofo grego, concebeu alguns princípios que, segundo ele, deveriam ser obedecidos por todas as formas de raciocínio legítimas. Hoje em dia, embora muitos sistemas lógicos obedeçam às leis aristotélicas, as mesmas não representam mais um consenso. Sem nos preocuparmos com o rigor, citaremos as três leis aristotélicas mais conhecidas.

(1) O **Princípio da Identidade** diz que uma proposição verdadeira sempre será verdadeira, e uma proposição falsa sempre será falsa.

(2) O **Princípio do Terceiro Excluído** diz que, dadas duas proposições contraditórias[1], pelo menos uma delas é verdadeira.

(3) O **Princípio da Não Contradição** diz que, entre duas proposições contraditórias, uma delas é falsa.

A partir do Século XX, especialmente após os anos 60, surgiu uma multiplicidade de sistemas lógicos. Neste trabalho abordaremos sistemas da lógica clássica nos níveis proposicional e quantificacional.

Um *sistema lógico* é definido em duas etapas. Na primeira é especificado o que vem a ser uma linguagem formal deste sistema. Para definirmos uma dada linguagem formal especificamos um alfabeto e uma gramática, para obtermos as expressões significativas do sistema. Na segunda etapa é feita uma distinção das fórmulas (relativamente) verdadeiras daquelas que não o são.

3.2 Linguagens para a Lógica Clássica

Uma *linguagem* é composta basicamente por um conjunto de sinais que servem para expressar uma idéia ou um sentimento. Mas a linguagem não é exclusiva do ser humano, podemos verificar que os mais variados animais também a possuem.

Uma linguagem pode ser classificada como *natural*[2] ou *artificial*. Uma *linguagem natural* é uma língua usada por alguma comunidade de seres humanos, em alguma época, para fins de comunicação, e, para descrevê-la, é necessária, em geral, uma grande gama de regras, pois a sua gramática é normalmente complexa. Já uma linguagem artificial ou formal

[1]Dizemos que duas proposições são contraditórias se uma delas é a negação da outra.

[2]São as línguas usadas corriqueiramente pelos seres humanos em sua comunicação.

possui uma gramática bem mais simples, cuja descrição pode ser feita, normalmente, em poucas linhas de texto. Tais linguagens, apesar de sua simplicidade, podem possuir um poder expressivo grande.

A *lógica simbólica* utiliza-se, para a sua expressão, das linguagens formais, pois as mesmas são mais adequadas, devido à sua inerente simplicidade.

Podemos construir uma *linguagem formal* em duas etapas. Primeiro escolhe-se um conjunto não vazio de sinais, chamado de *alfabeto*. Na segunda etapa faz-se uso de uma coleção de regras, em geral pequena, chamada de *gramática*. É a partir de um alfabeto que as expressões são formadas, entretanto a gramática é responsável em identificar dentre as expressões geradas quais são significativas. [13]

O *simbolismo* de uma linguagem formal utilizada dentro de um sistema lógico não constitui apenas uma prática taquigráfica, mas possui um grande poder de expressividade, comparável em muitos casos à versatilidade das linguagens naturais. Uma linguagem formal bem construída possui uma precisão absoluta quanto à construção de suas expressões significativas, no sentido de que os diversos tipos das mesmas não se sobrepõem, e cada uma é formada de uma única maneira. Esta precisão evita as tão conhecidas *irregularidades*, *ambigüidades* e *inconsistências* existentes nas linguagens naturais.

> "A motivação para o uso de tais linguagens é que os argumentos, originalmente apresentados em português, são traduzidos para uma linguagem cuja estrutura está precisamente especificada (ou seja, evitam-se os problemas de ambigüidade existentes nas linguagens naturais), uma linguagem na qual um argumento terá uma forma imediatamente reconhecível, e para a qual se pode dar uma definição precisa de conseqüência lógica." [38, pg. 62]

Uma linguagem pode ser abordada sob dois aspectos, o *sintático* e o *semântico*. Segundo [3], a principal diferença entre ambas está que a primeira abordagem se preocupa apenas com as expressões ou samblagens[3], de forma independente de qualquer interpretação. Já a semântica, ao contrário da sintaxe, se preocupa com os significados das expressões.

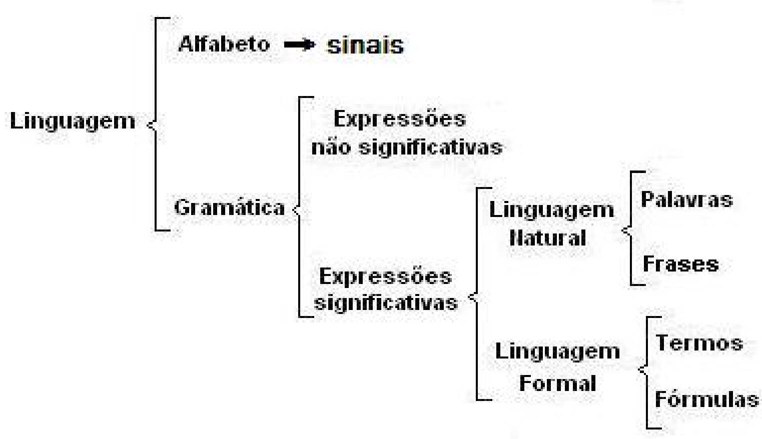


Figura 3.1: Estrutura de uma Linguagem

3.3 Semânticas de Valorações

A forma utilizada com mais freqüência pelos sistemas lógicos na definição de suas semânticas é a *semântica de valorações*. Uma *valoração* é uma função que associa cada fórmula de uma dada linguagem formal a um rótulo dito *valor veritativo*, atendendo a determinadas condições, as quais variam de lógica para lógica. Um *valor veritativo* é um grau de veracidade ou falsidade, o qual pode ser associado a uma dada fórmula por alguma valoração.

Uma semântica de valorações para uma lógica específica é dada por um *domínio veritativo*, que é uma coleção de valores veritativos, por um subconjunto desta coleção, cujos elementos são ditos *valores distinguidos*,

[3]Seqüência de caracteres, ou símbolos.

e, para cada linguagem desta lógica, uma *coleção de funções*, ditas *valorações* nesta lógica, que associam cada fórmula desta linguagem a um valor veritativo. Os valores veritativos *distinguidos* são os *graus de veracidade* desta semântica, enquanto que os demais valores veritativos, os *não distinguidos*, representam os *graus de falsidade* desta semântica.

3.3.1 Notação: Adotaremos nesta seção as seguintes convenções:
- $\mathsf{\mathbf{L}}$ é uma lógica dotada de uma semântica de valorações.
- **L**, $\mathbf{L}_1$ e $\mathbf{L}_2$ são linguagens para $\mathsf{\mathbf{L}}$.
- **P**, **Q**, **R** e **S** são fórmulas.
- Γ e ϕ são coleções de fórmulas.

3.3.2 Pré-definição: Consideramos como previamente definido o que é uma *$\mathsf{\mathbf{L}}$-valoração para* **L**.

3.3.3 Definição: Dizemos que ***V*** é *uma $\mathsf{\mathbf{L}}$-valoração* se existe uma linguagem **L** para $\mathsf{\mathbf{L}}$ tal que **V** é uma $\mathsf{\mathbf{L}}$-valoração para **L**.

3.3.4 Definição: Dizemos que ***V*** é uma *$\mathsf{\mathbf{L}}$-valoração para* **P** se existe uma linguagem **L** para $\mathsf{\mathbf{L}}$ tal que **P** é uma fórmula de **L** e **V** é uma $\mathsf{\mathbf{L}}$-valoração para **L**. Dizemos também que **V** é uma $\mathsf{\mathbf{L}}$-valoração para Γ se existe uma linguagem **L** para $\mathsf{\mathbf{L}}$ tal que Γ é um coleção de fórmulas de **L** e **V** é uma $\mathsf{\mathbf{L}}$-valoração para **L**.

3.3.5 Definição: Dizemos que *uma $\mathsf{\mathbf{L}}$-valoração* **V** *satisfaz* **P** se **V** é uma $\mathsf{\mathbf{L}}$-valoração para **P** que associa **P** a um valor distinguido em $\mathsf{\mathbf{L}}$. Dizemos também que uma $\mathsf{\mathbf{L}}$-valoração **V** satisfaz Γ se **V** satisfizer todas as fórmulas de Γ.

3.3.6 Definição:
- **P** é *$\mathsf{\mathbf{L}}$-satisfatível* $\equiv$ existe uma $\mathsf{\mathbf{L}}$-valoração que satisfaz **P**.
- Γ é *$\mathsf{\mathbf{L}}$-satisfatível* $\equiv$ existe uma $\mathsf{\mathbf{L}}$-valoração que satisfaz Γ.
- **P** é *$\mathsf{\mathbf{L}}$-insatisfatível* $\equiv$ nenhuma $\mathsf{\mathbf{L}}$-valoração satisfaz **P**.
- Γ é *$\mathsf{\mathbf{L}}$-insatisfatível* $\equiv$ nenhuma $\mathsf{\mathbf{L}}$-valoração satisfaz Γ.
- **P** é *$\mathsf{\mathbf{L}}$-válido* $\equiv$ toda $\mathsf{\mathbf{L}}$-valoração para **P** satisfaz **P**.
- **P** é *$\mathsf{\mathbf{L}}$-inválido* $\equiv$ alguma $\mathsf{\mathbf{L}}$-valoração para **P** não satisfaz **P**.
- Γ é *$\mathsf{\mathbf{L}}$-válido* $\equiv$ toda $\mathsf{\mathbf{L}}$-valoração para Γ satisfaz alguma fórmula de Γ.
- Γ é *$\mathsf{\mathbf{L}}$-inválido* $\equiv$ alguma $\mathsf{\mathbf{L}}$-valoração para Γ não satisfaz nenhuma fórmula de Γ.

3.3.7 Escólio: As seguintes proposições são válidas:
- $\{\mathbf{P}\}$ é **L**-válido se, e somente se, **P** é **L**-válido.
- $\{\mathbf{P}\}$ é **L**-satisfatível se, e somente se, **P** é **L**-satisfatível.

3.3.8 Escólio: As seguintes proposições são válidas:
- **P** é **L**-insatisfatível se, e somente se, não é verdade que **P** é **L**-satisfatível.
- Γ é **L**-insatisfatível se, e somente se, não é verdade que Γ é **L**-satisfatível.
- **P** é **L**-inválido se, e somente se, não é verdade que **P** é **L**-válido.
- Γ é **L**-inválido se, e somente se, não é verdade que Γ é **L**-válido.

No caso da lógica proposicional clássica construímos as suas valorações simplesmente dando o seu efeito sobre as letras sentenciais. Na maioria das lógicas dotadas de semânticas de valorações, incluindo algumas lógicas proposicionais não clássicas, não é possível construir suas valorações com a mesma facilidade. Para muitas destas é necessário definir antes um objeto intermediário, denominado *interpretação*. Uma interpretação dá significados a todos os sinais da linguagem adotada.

3.3.9 Pré-definição: Consideramos como previamente definido o que é ***L**-interpretação para* **L**. Dada uma **L**-interpretação para **L**, consideramos também previamente definida a função $\mathbf{I}_V$, da coleção de fórmulas em **L** para o domínio veritativo de **L**. A coleção de todas as ***L**-valorações para* **L** é o conjunto de todas as funções $\mathbf{I}_V$, onde **I** é uma **L**-interpretação para **L**.

3.3.10 Pré-definição: Se **L** é uma lógica cujas linguagens admitem termos, e se **I** é uma **L**-interpretação para **L**, consideraremos como previamente definido o universo de **I**, o qual é um conjunto não vazio. Consideraremos como também previamente definida a função $\boldsymbol{I}_D$ da coleção dos termos em **L** para o universo de **I**, dita a ***L**-denotação especificada por* **I**.

3.3.11 Definição: **I** é *uma **L**-interpretação* se existe uma linguagem **L** para **L** tal que **I** é uma **L**-interpretação para **L**.

3.3.12 Definição: Se **L** é uma lógica cujas linguagens possuem termos, dizemos que **D** é um *designador em* **L** se **D** é um termo em **L** ou **D** é uma fórmula em **L**. Uma **L**-interpretação **I** é dita uma **L**-interpretação para um designador **D** se uma das duas condições seguintes forem satisfeitas:
- **D** é um termo em **L** e **I** é uma **L**-interpretação para **L**.

- **D** é uma fórmula de **L** e **I** é uma **L**-interpretação para **L**.

3.3.13 Definição: Dizemos que uma **L**-interpretação **I** é uma **L**-interpretação para uma coleção Γ de designadores em alguma linguagem para **L** se, para todo $\mathbf{D} \in \Gamma$, **I** é uma **L**-interpretação para **D**.

3.3.14 Definição: Dizemos que *uma* ***L****-interpretação* **I** *satisfaz* **P** se a **L**-valoração $\mathbf{I}_V$ satisfaz **P**. Dizemos também que uma **L**-interpretação **I** satisfaz Γ se a **L**-valoração $\mathbf{I}_V$ satisfaz Γ.

3.3.15 Definição: As seguintes proposições são válidas:
- **P** é ***L****-satisfatível* se, e somente se, existir uma **L**-interpretação que satisfaz **P**.
- Γ é ***L****-satisfatível* se, e somente se, existir uma **L**-interpretação que satisfaz Γ.
- **P** é ***L****-insatisfatível* se, e somente se, nenhuma **L**-interpretação satisfizer **P**.
- Γ é ***L****-insatisfatível* se, e somente se, nenhuma **L**-interpretação satisfizer Γ.
- **P** é ***L****-válido* se, e somente se, toda **L**-interpretação para **P** satisfaz **P**.
- **P** é ***L****-inválido* se, e somente se, alguma **L**-interpretação para **P** não satisfaz **P**.
- Γ é ***L****-válido* se, e somente se, toda **L**-interpretação para Γ satisfaz alguma fórmula de Γ.
- Γ é ***L****-inválido* se, e somente se, alguma **L**-interpretação para Γ não satisfaz nenhuma fórmula de Γ.
- $\Gamma \models_{\mathbf{L}} \mathbf{P}$ se, somente se, toda **L**-interpretação para $\Gamma \cup \{\mathbf{P}\}$ que satisfaz Γ satisfaz **P**.

A *semântica clássica*, associada à lógica clássica em qualquer um de seus níveis, seja proposicional, quantificacional ou equacional, possui um domínio veritativo com apenas dois valores, e um único valor distinguido, isto é, tal semântica possui um único grau de veracidade e um único grau de falsidade.

3.3.16 Definição: A semântica clássica possui dois valores veritativos, atendendo às seguintes condições:
- "**v**" é o único valor distinguido e pode ser lido como "verdadeiro".
- "**f**" é o único valor não distinguido e pode ser lido como "falso".

3.4 Linguagens Proposicionais

A definição de uma *linguagem proposicional* é o primeiro passo para podermos fixar um sistema formal para a lógica proposicional clássica.

3.4.1 Definição: Um *alfabeto proposicional* é uma coleção de símbolos constituída por três tipos de sinais, mutuamente disjuntos, ditos *letras sentenciais*, *conectivos* e *sinais de pontuação*, onde as seguintes condições são satisfeitas:
- As coleções de letras sentenciais e de conectivos são não vazias;
- Os sinais de pontuação são "(", o parêntese de abertura, e ")", o parêntese de fechamento.

A coleção de conectivos pode variar, dependendo do sistema lógico considerado.

3.4.2 Definição: (Alfabeto para a Lógica Proposicional Clássica)
Um *alfabeto para a lógica proposicional clássica* é uma alfabeto proposicional, no qual a coleção de conectivos é formada pelos sinais "$\neg$", "$\rightarrow$", "$\wedge$" e "$\vee$".

3.4.3 Notação: Φ denota um alfabeto para a lógica proposicional clássica e $\mathbf{L}_\Phi$ é a linguagem proposicional construída a partir de Φ.

Os *conectivos* são sinais que formam fórmulas a partir de fórmulas. Cada conectivo é dotado de uma aridade, a qual é o número de fórmulas necessárias para formação de uma nova fórmula. Em $\mathbf{L}_\Phi$ o conectivo "$\neg$" possui aridade um, ou seja, é *monádico*, enquanto que os demais conectivos em $\mathbf{L}_\Phi$ possuem aridade dois, ou seja, são *diádicos*.

3.4.4 Definição: (Fórmula de $\mathbf{L}_\Phi$)
As seguintes cláusulas definem as *regras de formação* de fórmulas de $\mathbf{L}_\Phi$:
- Todas as letras sentenciais de Φ são *fórmulas* de $\mathbf{L}_\Phi$, ditas *fórmulas atômicas*.
- Se $\mathbf{P}$ é uma fórmula de $\mathbf{L}_\Phi$, então $\neg\mathbf{P}$ também é uma fórmula de $\mathbf{L}_\Phi$, denominada a *negação de* $\mathbf{P}$.
- Se $\mathbf{P}$ e $\mathbf{Q}$ são fórmulas de $\mathbf{L}_\Phi$, então:
  - ($\mathbf{P} \rightarrow \mathbf{Q}$) é uma fórmula em $\mathbf{L}_\Phi$, denominada *implicação*, onde $\mathbf{P}$ é dito o seu *antecedente* e $\mathbf{Q}$ é dito o seu *conseqüente*.

- ○ ($\mathbf{P} \wedge \mathbf{Q}$) é uma fórmula de $\mathbf{L}_\Phi$, denominada *conjunção*, onde **P** e **Q** são ditos os seus *conjuntores*.
- ○ ($\mathbf{P} \vee \mathbf{Q}$) é uma fórmula de $\mathbf{L}_\Phi$, denominada *disjunção*, onde **P** e **Q** são ditos os seus *disjuntores*.

Com respeito à lógica proposicional clássica o conectivo “$\neg$” é monádico, enquanto que os conectivos “$\rightarrow$”, “$\wedge$” e “$\vee$” são diádicos.

3.4.5 Definição: (Equivalência de Fórmulas)
$(\mathbf{P} \leftrightarrow \mathbf{Q}) \equiv ((\mathbf{P} \rightarrow \mathbf{Q}) \wedge (\mathbf{Q} \rightarrow \mathbf{P}))$.
A fórmula “$(\mathbf{P} \leftrightarrow \mathbf{Q})$” é denominada *equivalência de* **P** *e* **Q**, onde **P** e **Q** são ditos os seus *membros*.

3.4.6 Convenções: Para diminuir a necessidade do uso de parênteses, valem as seguintes normas para escrita informal de fórmulas:

- Quando as fórmulas $(\mathbf{P} \rightarrow \mathbf{Q})$, $(\mathbf{P} \wedge \mathbf{Q})$ e $(\mathbf{P} \vee \mathbf{Q})$ não estiverem inseridas dentro de uma fórmula maior, pode-se dispensar o uso dos parênteses externos.
- Cada conectivo diádico possui uma prioridade de agrupamento em relação aos outros. A ordem de prioridade é dada por $\{\wedge, \vee\}$, $\rightarrow$, $\leftrightarrow$. Por exemplo: “$\mathbf{Q} \leftrightarrow \mathbf{S} \wedge \mathbf{P} \rightarrow \mathbf{S}$” representa “$\mathbf{Q} \leftrightarrow ((\mathbf{S} \wedge \mathbf{P}) \rightarrow \mathbf{S})$”.
- Quando o sinal de implicação se suceder, os parênteses são inseridos da direita para a esquerda. Por exemplo: “$\mathbf{Q} \rightarrow \mathbf{R} \rightarrow \mathbf{S} \rightarrow \mathbf{P} \rightarrow \mathbf{S}$” representa “$\mathbf{Q} \rightarrow (\mathbf{R} \rightarrow (\mathbf{S} \rightarrow (\mathbf{P} \rightarrow \mathbf{S})))$”.
- Quando um conectivo diádico distinto do sinal de implicação se suceder, os parênteses são inseridos da esquerda para direita. Por exemplo: “$\mathbf{P} \wedge \mathbf{Q} \wedge \mathbf{R} \wedge \mathbf{S}$” representa “$((\mathbf{P} \wedge \mathbf{Q}) \wedge \mathbf{R}) \wedge \mathbf{S}$”.

3.5 Uma Semântica para a Lógica Proposicional Clássica

Definiremos nesta seção uma semântica para a *lógica proposicional clássica*, a qual denotamos, de agora em diante, pela sigla “**LPC**”.

3.5.1 Definição: Dotamos o conjunto $\{\mathbf{v}, \mathbf{f}\}$ de uma relação de ordem estrita, considerando **f** menor que **v**.

3.5.2 Definição: (LPC-valoração)
Uma ***LPC**-valoração* para $\mathbf{L}_\Phi$ é uma função **V** de $\mathbf{L}_\Phi$ em $\{\mathbf{v}, \mathbf{f}\}$, obedecendo às seguintes condições:

- $\mathbf{V}(\neg \mathbf{P}) \neq \mathbf{V}(\mathbf{P})$;
- $\mathbf{V}(\mathbf{P} \rightarrow \mathbf{Q}) = \max\{\mathbf{V}(\neg \mathbf{P}), \mathbf{V}(\mathbf{Q})\}$;
- $\mathbf{V}(\mathbf{P} \wedge \mathbf{Q}) = \min\{\mathbf{V}(\mathbf{P}), \mathbf{V}(\mathbf{Q})\}$;
- $\mathbf{V}(\mathbf{P} \vee \mathbf{Q}) = \max\{\mathbf{V}(\mathbf{P}), \mathbf{V}(\mathbf{Q})\}$.

3.5.3 Escólio: Uma **LPC**-valoração é determinada a partir do seu efeito sobre as letras sentenciais.

3.5.4 Escólio: Uma função de $\mathbf{L}_\Phi$ em $\{\mathbf{v}, \mathbf{f}\}$ é uma **LPC**-valoração se, e somente se, as seguintes condições forem satisfeitas:

- $\mathbf{V}(\neg \mathbf{P}) \neq \mathbf{V}(\mathbf{P})$;
- $\mathbf{V}(\mathbf{P} \rightarrow \mathbf{Q}) = \mathbf{v}$ se, somente se, $\mathbf{V}(\mathbf{P}) = \mathbf{f}$ ou $\mathbf{V}(\mathbf{Q}) = \mathbf{v}$;
- $\mathbf{V}(\mathbf{P} \wedge \mathbf{Q}) = \mathbf{v}$ se, somente se, $\mathbf{V}(\mathbf{P}) = \mathbf{v}$ e $\mathbf{V}(\mathbf{Q}) = \mathbf{v}$;
- $\mathbf{V}(\mathbf{P} \vee \mathbf{Q}) = \mathbf{v}$ se, somente se, $\mathbf{V}(\mathbf{P}) = \mathbf{v}$ ou $\mathbf{V}(\mathbf{Q}) = \mathbf{v}$.

3.6 Linguagens Quantificacionais

As *linguagens quantificacionais* englobam todas as características das linguagens proposicionais, e possuem em seus alfabetos novos símbolos, utilizados para uma análise interna das fórmulas e para a expressão de quantificações.

3.6.1 Definição: (Alfabeto para a Lógica Quantificacional Clássica)
Um *alfabeto para a lógica quantificacional clássica* possui os seguintes tipos de sinais, mutuamente disjuntos dois a dois:

- uma coleção infinita enumerável de sinais ditos *variáveis*, a qual é a mesma em todos os alfabetos quantificacionais;
- a coleção dos sinais "$\neg$","$\wedge$","$\vee$" e "$\rightarrow$", ditos *conectivos*;
- a coleção dos sinais "$\forall$" e "$\exists$", ditos *quantificadores*;
- os sinais de pontuação "(", o parêntese de abertura, ")", o parêntese de fechamento, e ",", a vírgula;
- uma coleção vazia ou não de sinais ditos *constantes*;
- uma coleção vazia ou não de sinais ditos *sinais funcionais*, na qual cada sinal funcional é associado a uma coleção não vazia de números naturais positivos, chamados de suas *aridades*;
- uma coleção não vazia de sinais ditos *sinais predicativos*, na qual cada sinal predicativo é associado a uma coleção não vazia de números naturais positivos, chamados de suas *aridades*.

3.6.2 Notação: Φ^* denota um alfabeto para a lógica quantificacional clássica, e $\mathbf{L}_{\Phi^*}$ é a linguagem quantificacional gerada por Φ^*.

Consideramos aqui, para fins de simplicidade, que a função sintática de um sinal usado em diferentes alfabetos da mesma lógica permanece inalterada. Por exemplo, se considerarmos "**p**" como um sinal predicativo de aridade 2, ficará implícito que este sinal possui esta mesma função em qualquer linguagem da mesma lógica, cujo alfabeto contenha este sinal [23].

3.6.3 Notação: Usaremos as seguintes convenções para os grupos de letras abaixo, seguidas ou não de plicas ou subíndices:

- ***x***, ***y***, ***z*** e ***w*** - variáveis.
- **a**, **b** e **c** - constantes.
- **f**, **g** e **h** - sinais funcionais.
- **p**, **q** e **r** - sinais predicativos.
- **t**, **u**, **v** - termos em $\mathbf{L}_{\Phi^*}$.
- **P**, **Q**, **R** e **S** - fórmulas em $\mathbf{L}_{\Phi^*}$.

3.6.4 Definição: (Termos e Fórmulas em $\mathbf{L}_{\Phi^*}$)
As seguintes cláusulas definem as *regras de formação* de termos e fórmulas em $\mathbf{L}_{\Phi^*}$:

- Toda variável é um termo (atômico) em $\mathbf{L}_{\Phi^*}$.
- Toda constante em $\mathbf{L}_{\Phi^*}$ é um termo (atômico) em $\mathbf{L}_{\Phi^*}$.
- Se $\mathbf{t}_1,...,\mathbf{t}_n$ são **n** termos em $\mathbf{L}_{\Phi^*}$ e **f** é um sinal funcional de Φ^* de aridade **n**, então $\mathbf{f}(\mathbf{t}_1,...,\mathbf{t}_n)$ é um termo em $\mathbf{L}_{\Phi^*}$, dito *termo funcional* em $\mathbf{L}_{\Phi^*}$.
- Se $\mathbf{t}_1,...,\mathbf{t}_n$ são **n** termos em $\mathbf{L}_{\Phi^*}$ e **p** é um sinal predicativo de Φ^* de aridade **n**, então $\mathbf{p}(\mathbf{t}_1,...,\mathbf{t}_n)$ é uma *fórmula atômica* em $\mathbf{L}_{\Phi^*}$.
- Se **P** é uma fórmula em $\mathbf{L}_{\Phi^*}$, então $\neg\mathbf{P}$ também o é.
- Se **P** e **Q** são fórmulas em $\mathbf{L}_{\Phi^*}$, então $(\mathbf{P} \rightarrow \mathbf{Q})$, $(\mathbf{P} \wedge \mathbf{Q})$, $(\mathbf{P} \vee \mathbf{Q})$ e $(\mathbf{P} \wedge \mathbf{Q})$, também o são.
- Se ***x*** é uma variável e **P** uma fórmula em $\mathbf{L}_{\Phi^*}$, então $\forall \boldsymbol{x}\, \mathbf{P}$ e $\exists \boldsymbol{x}\, \mathbf{P}$ são fórmulas em $\mathbf{L}_{\Phi^*}$, denominadas respectivamente *fórmula universal* e *fórmula existencial*.

Os símbolos **P** e **Q** usados na definição 3.6.4 não fazem parte da linguagem $\mathbf{L}_{\Phi^*}$. Os mesmos são *variáveis metalingüísticas* (ou *variáveis sintáticas*) para fórmulas em $\mathbf{L}_{\Phi^*}$, no sentindo de que qualquer resultado em que tais variáveis sejam usadas diz respeito a fórmulas arbitrárias de $\mathbf{L}_{\Phi^*}$. Da mesma forma os sinais **t**, **u**, **v** são metavariáveis para termos em $\mathbf{L}_{\Phi^*}$.

Fazendo uma analogia com as linguagens naturais, os termos são os nomes e pronomes. Semanticamente os mesmos representam elementos do universo de discurso, que é a coleção de objetos de estudo em um dado momento [21].

3.6.5 Definição: (Subfórmula de uma dada fórmula de $\mathbf{L}_{\Phi^*}$)
As seguintes cláusulas definem o que se entende por *subfórmula de* uma dada fórmula:

- **P** é subfórmula de **P**.
- **P** é subfórmula de $\neg$**P**.
- **P** e **Q** são subfórmulas de (**P** $\rightarrow$ **Q**), (**P** $\wedge$ **Q**) e (**P** $\vee$ **Q**).
- **P** é subfórmula de $\forall \boldsymbol{x}\, \mathbf{P}$ e de $\exists \boldsymbol{x}\, \mathbf{P}$.
- Se **P** é subfórmula de **Q** e **Q** é subfórmula de **R**, então **P** é subfórmula de **R**.

3.6.6 Definição: Uma *ocorrência de uma variável* ***x*** em uma fórmula **P** é dita *ligada* em **P** se esta ocorrência suceder em **P** a um dos sinais "$\forall$" ou "$\exists$", ou se **P** possuir uma subfórmula de uma das formas $\forall \boldsymbol{x}\, \mathbf{Q}$ ou $\exists \boldsymbol{x}\, \mathbf{Q}$ e esta ocorrência figurar em **Q**.

3.6.7 Definição: Uma *ocorrência de* ***x*** é dita ser *livre em* **P** se esta ocorrência não for ligada em **P**.

3.6.8 Definição: Uma variável ***x*** é dita ser *ligada em* **P** se ***x*** possuir pelo menos uma ocorrência ligada em **P**.

3.6.9 Definição: Uma variável ***x*** é dita ser *livre em* **P** se ***x*** possuir pelo menos uma ocorrência livre em **P**[4].

Segundo [13] toda ocorrência livre de uma dada variável ***x*** em uma fórmula **P** está citando naquele ponto o objeto do universo de discurso de-

[4]Uma dada variável pode ser ao mesmo tempo livre e ligada em uma fórmula, caso esta possua tanto uma ocorrência livre como uma ocorrência ligada nesta fórmula.

signado por $\boldsymbol{x}$. Já uma ocorrência ligada de $\boldsymbol{x}$ em $\mathbf{P}$ não tem nada a ver com o objeto designado por $\boldsymbol{x}$, mas sim com a expressão de uma quantificação.

3.6.10 Escólio: (Variável Livre em uma Fórmula)
Definimos as seguintes regras para determinar quando uma variável é *livre* em uma *fórmula*:

- Se $\mathbf{P}$ é uma fórmula atômica, então $\boldsymbol{x}$ é livre em $\mathbf{P}$ se, e somente se, $\boldsymbol{x}$ possuir pelo menos uma ocorrência em $\mathbf{P}$.
- $\boldsymbol{x}$ é livre em $\neg\mathbf{P}$ se, e somente se, $\boldsymbol{x}$ for livre em $\mathbf{P}$.
- $\boldsymbol{x}$ é livre em qualquer uma das fórmulas $(\mathbf{P} \rightarrow \mathbf{Q})$, $(\mathbf{P} \wedge \mathbf{Q})$ ou $(\mathbf{P} \vee \mathbf{Q})$ se, e somente se, $\boldsymbol{x}$ for livre em $\mathbf{P}$ ou $\boldsymbol{x}$ for livre em $\mathbf{Q}$.
- $\boldsymbol{x}$ é livre em qualquer uma das fórmulas $\forall \boldsymbol{y}\, \mathbf{P}$ ou $\exists \boldsymbol{y}\, \mathbf{P}$ se, e somente se, $\boldsymbol{x}$ for distinto de $\boldsymbol{y}$ e $\boldsymbol{x}$ for livre em $\mathbf{P}$.
- $\boldsymbol{x}$ não é livre nas fórmulas $\forall \boldsymbol{x}\, \mathbf{P}$ ou $\exists \boldsymbol{x}\, \mathbf{P}$.

3.6.11 Definição: (Termo Fechado)
Termos que não contêm variáveis livres são ditos *termos fechados*.

3.6.12 Definição: (Sentença)
Fórmulas que não contém variáveis livres são chamadas de *sentenças*.

Iremos definir em seguida *instanciação de variáveis por* termos *em* fórmulas. Usamos instanciação no decurso da introdução ou eliminação de quantificadores nos processos de raciocínio da lógica quantificacional.

3.6.13 Definição: (Instanciação)

- *A instanciação de* $\boldsymbol{x}$ *por* $\mathbf{t}$ *em* $\mathbf{u}$, notada por $\mathbf{u}(\boldsymbol{x}|\mathbf{t})$, é o termo obtido de $\mathbf{u}$ substituindo todas as ocorrências de $\boldsymbol{x}$ por $\mathbf{t}$.
- *A instanciação de* $\boldsymbol{x}$ *por* $\mathbf{t}$ *em* $\mathbf{P}$, notada por $\mathbf{P}(\boldsymbol{x}|\mathbf{t})$, é a fórmula obtida de $\mathbf{P}$ substituindo todas as ocorrências livres de $\boldsymbol{x}$ por $\mathbf{t}$.

3.6.14 Definição: (Instanciação Simultânea)

- A *instanciação simultânea de* $\boldsymbol{x}_1,...,\boldsymbol{x}_n$ por $\mathbf{t}_1,...,\mathbf{t}_n$ *em* $\mathbf{u}$, notada por $\mathbf{u}(\boldsymbol{x}_1,...,\boldsymbol{x}_n|\mathbf{t}_1,...,\mathbf{t}_n)$, é o termo obtido de $\mathbf{u}$ substituindo simultaneamente todas as ocorrências de $\boldsymbol{x}_1,...,\boldsymbol{x}_n$, respectivamente, por $\mathbf{t}_1,...,\mathbf{t}_n$.
- A *instanciação simultânea de* $\boldsymbol{x}_1,...,\boldsymbol{x}_n$ por $\mathbf{t}_1,...,\mathbf{t}_n$ *para* a fórmula $\mathbf{P}$, notada por $\mathbf{P}(\boldsymbol{x}_1,...,\boldsymbol{x}_n|\mathbf{t}_1,...,\mathbf{t}_n)$, é a fórmula obtida de $\mathbf{P}$ substituindo simultaneamente todas as ocorrências de $\boldsymbol{x}_1,...,\boldsymbol{x}_n$, respectivamente, por $\mathbf{t}_1,...,\mathbf{t}_n$.

3.7 Uma Semântica para a Lógica Quantificacional Clássica

Para atribuir significados às fórmulas de uma linguagem quantificacional é necessário definir os valores semânticos das constantes, dos sinais funcionais, dos sinais predicativos e das variáveis. Uma interpretação define duas funções semânticas básicas, a *denotação* e a *valoração*. A primeira associa termos a indivíduos do universo de discurso da interpretação, e a segunda associa fórmulas à valores veritativos.

Denotaremos a *lógica quantificacional clássica*, de agora em diante, pela sigla "**LQC**".

3.7.1 Notação: Δ é um conjunto não vazio, e **d** $\in \Delta$.

3.7.2 Definição: Um *domínio formal* é uma coleção **D** cujos elementos são de uma das formas **c**, $\langle$**n**,**f**$\rangle$ ou $\langle$**n**,**p**$\rangle$, onde **c** é uma constante, **f** é um sinal funcional, **p** é um sinal predicativo e **n** é um número natural, satisfazendo às seguintes condições:

- Se $\langle$**n**,**f**$\rangle \in$ **D**, então **n** $\neq 0$.
- Existe pelo menos um sinal predicativo **p** e um número natural **n** tal que $\langle$**n**,**p**$\rangle \in$ **D**.

3.7.3 Definição: Um *mundo sobre* Δ é uma função σ cujo domínio é um domínio formal, atendendo às seguintes condições:

- Para cada constante **c** $\in$ D(σ), σ(**c**) $\in \Delta$.
- Para cada sinal funcional **f** de aridade **n**, se $\langle$**n**,**f**$\rangle \in$ D(σ), então σ(**n**,**f**) é uma função de Δ^n em Δ.
- Para cada sinal predicativo **p** de aridade **n**, se $\langle$**n**,**p**$\rangle \in$ D(σ), então σ(**n**,**p**) é um subconjunto de Δ^n.

3.7.4 Notação: σ é um mundo sobre Δ.

3.7.5 Definição: Um *mundo sobre* Δ *para* $\mathbf{L}_{\Phi^*}$ é um mundo sobre Δ cujo domínio atende às seguintes condições:

- Para cada constante **c** de $\mathbf{L}_{\Phi^*}$, **c** $\in$ D(σ).
- Para cada sinal funcional **n**-ário **f** de $\mathbf{L}_{\Phi^*}$, $\langle$**n**,**f**$\rangle \in$ D(σ).

- Para cada sinal predicativo **n**-ário **p** de $\mathbf{L}_{\Phi^*}$, $\langle \mathbf{n},\mathbf{p}\rangle \in \mathrm{D}(\sigma)$.

3.7.6 Definição: Dizemos que σ é um *mundo* se existir Δ tal que $\Delta \neq \emptyset$ e σ é um mundo sobre Δ.

3.7.7 Definição: Dizemos que **s** é uma Δ-*atribuição para variáveis* se **s** é uma função da coleção de variáveis para Δ.

3.7.8 Definição: Uma ***LQC**-interpretação* (para $\mathbf{L}_{\Phi^*}$) é uma tripla $\mathbf{I}=\langle\Delta,\sigma,\mathbf{s}\rangle$, onde Δ é um conjunto não vazio, dito o *universo* de **I**, σ é um mundo sobre Δ (para $\mathbf{L}_{\Phi^*}$), dito o *mundo de* **I**, e **s** é uma Δ-atribuição para variáveis, dita *a atribuição para variáveis de* **I**.

3.7.9 Definição: Dada uma Δ-*atribuição para* variáveis **s** e um elemento $\mathbf{d} \in \Delta$, definimos uma nova Δ-atribuição de variáveis, notada por $\mathbf{s}(\boldsymbol{x}|\mathbf{d})$, segundo a cláusula abaixo:

- $\mathbf{s}(\boldsymbol{x}|\mathbf{d})(\boldsymbol{y}) = \begin{cases} \mathbf{d}, \text{ se } \boldsymbol{y} = \boldsymbol{x}, \\ \mathbf{s}(\boldsymbol{y}), \text{ se } \boldsymbol{y} \neq \boldsymbol{x}. \end{cases}$

3.7.10 Definição: Se $\mathbf{I} = \langle\Delta, \sigma,\mathbf{s}\rangle$ é uma **LQC**-interpretação para $\mathbf{L}_{\Phi^*}$ e $\mathbf{d} \in \Delta$, então $\mathbf{I}(\boldsymbol{x}|\mathbf{d})$ é uma **LQC**-interpretação notada por $\mathbf{I} = \langle\Delta, \sigma,\mathbf{s}(\boldsymbol{x}|\mathbf{d})\rangle$.

3.7.11 Definição: (Denotação e Valoração)
Dada uma **LQC**-interpretação $\mathbf{I} = \langle\Delta, \sigma, \mathbf{s}\rangle$ para $\mathbf{L}_{\Phi^*}$, definimos duas funções $\mathbf{I}_V$ e $\mathbf{I}_D$, ditas respectivamente *a **LQC**-valoração* e *a denotação geradas por* **I**, obedecendo às seguintes cláusulas:

- $\mathbf{I}_D$ é uma função que associa cada termo em $\mathbf{L}_{\Phi^*}$ a um elemento de Δ;
- $\mathbf{I}_D(\mathbf{c}) = \sigma(\mathbf{c})$;
- $\mathbf{I}_D(\boldsymbol{x}) = \mathbf{s}(\boldsymbol{x})$;
- $\mathbf{I}_D(\mathbf{f}(\mathbf{t}_1,...,\mathbf{t}_n)) = \sigma(\mathbf{n},\mathbf{f})(\mathbf{I}_D(\mathbf{t}_1),...,\mathbf{I}_D(\mathbf{t}_n))$;
- $\mathbf{I}_V$ é uma função que associa cada fórmula em $\mathbf{L}_{\Phi^*}$ a um dos valores veritativos **v** ou **f**;
- $\mathbf{I}_V(\mathbf{p}(\mathbf{t}_1,...,\mathbf{t}_n)) = \mathbf{v}$ se, somente se, $\langle\mathbf{I}_D(\mathbf{t}_1),..., \mathbf{I}_D(\mathbf{t}_n)\rangle \in \sigma(\mathbf{n},\mathbf{p})$;
- $\mathbf{I}_V(\mathbf{P} \to \mathbf{Q}) = \begin{cases} \mathbf{v}, \text{ se } \mathbf{I}_V(\mathbf{P}) = \mathbf{f}; \\ \mathbf{I}_V(\mathbf{Q}), \text{ se } \mathbf{I}_V(\mathbf{P}) = \mathbf{v}; \end{cases}$
- $\mathbf{I}_V(\mathbf{P} \wedge \mathbf{Q}) = \min\{\mathbf{I}_V(\mathbf{P}), \mathbf{I}_V(\mathbf{Q})\}$;
- $\mathbf{I}_V(\mathbf{P} \vee \mathbf{Q}) = \max\{\mathbf{I}_V(\mathbf{P}), \mathbf{I}_V(\mathbf{Q})\}$;
- $\mathbf{I}_V(\neg\mathbf{P}) \neq \mathbf{I}_V(\mathbf{P})$;
- $\mathbf{I}_V(\forall\boldsymbol{x}\,\mathbf{P}) = \min\{\mathbf{I}(\boldsymbol{x}|\mathbf{d}))_V(\mathbf{P}) \mid \mathbf{d} \in \Delta\}$;
- $\mathbf{I}_V(\exists\boldsymbol{x}\,\mathbf{P}) = \max\{\mathbf{I}(\boldsymbol{x}|\mathbf{d}))_V(\mathbf{P}) \mid \mathbf{d} \in \Delta\}$.

3.7.12 Definição: Dizemos que $\boldsymbol{x}$ *aceita* **t** *em* **P** se não existir uma variável $\boldsymbol{y}$ ocorrendo em **t** e uma ocorrência livre de $\boldsymbol{x}$ em **P**, tal que a mesma figure em uma subfórmula de **P** de uma das formas $\forall \boldsymbol{y}\, \mathbf{Q}$ ou $\exists \boldsymbol{y}\, \mathbf{Q}$.

3.7.13 Lema: (**Significado da Instanciação de Variáveis por Termos**)
Seja $\mathbf{I} = \langle \Delta, \sigma, \mathbf{s} \rangle$ uma **LQC**-interpretação.

- Se **I** é uma **LQC**-interpretação para $\{\mathbf{u}, \mathbf{t}\}$, então $\mathbf{I}_D\mathbf{u}(\boldsymbol{x}|\mathbf{t}) = \mathbf{I}(\boldsymbol{x}|\mathbf{I}_D(\mathbf{t}))_D(\mathbf{u})$.
- Se **I** é uma **LQC**-interpretação para $\{\mathbf{P}, \mathbf{t}\}$ e $\boldsymbol{x}$ aceita **t** em **P**, então $\mathbf{I}_V(\mathbf{P}(\boldsymbol{x}|\mathbf{t})) = \mathbf{I}(\boldsymbol{x}|\mathbf{I}_D(\mathbf{t}))_V(\mathbf{P})$.

3.7.14 Definição: Sejam $\mathbf{I}=\langle \Delta,\sigma,\mathbf{s} \rangle$ e $\mathbf{I}'=\langle \Delta',\sigma',\mathbf{s}' \rangle$ duas **LQC**-interpretações.

Se **I** e **I'** são **LQC**-interpretações para um dado termo **t**, dizemos que **I** e **I'** coincidem em **t** se as seguintes condições forem satisfeitas:

- Se $\boldsymbol{x}$ ocorre em **t**, então $\mathbf{s}(\boldsymbol{x}) = \mathbf{s}'(\boldsymbol{x})$.
- Se **c** é uma constante figurando em **t**, então $\mathbf{c} \in D(\sigma) \cap D(\sigma')$ e $\sigma(\mathbf{c}) = \sigma'(\mathbf{c})$.
- Se **f** é um sinal funcional **n**-ário figurando em **t**, então $\langle \mathbf{n}, \mathbf{f} \rangle \in D(\sigma) \cap D(\sigma')$ e $\sigma(\mathbf{n}, \mathbf{f}) = \sigma'(\mathbf{n}, \mathbf{f})$.

Se **I** e **I'** são **LQC**-interpretações para uma fórmula **P**, dizemos que **I** e **I'** coincidem em **P** se as seguintes condições forem satisfeitas:

- Se $\boldsymbol{x}$ é livre em **P**, então $\mathbf{s}(\boldsymbol{x}) = \mathbf{s}'(\boldsymbol{x})$.
- Se **c** é uma constante figurando em **P**, então $\mathbf{c} \in D(\sigma) \cap D(\sigma')$ e $\sigma(\mathbf{c}) = \sigma'(\mathbf{c})$.
- Se **f** é um sinal funcional **n**-ário figurando em **f**, então $\langle \mathbf{n},\mathbf{f} \rangle \in D(\sigma) \cap D(\sigma')$ e $\sigma(\mathbf{n},\mathbf{f}) = \sigma'(\mathbf{n},\mathbf{f})$.
- Se **p** é um sinal predicativo **n**-ário figurando em **P**, então $\langle \mathbf{n},\mathbf{p} \rangle \in D(\sigma) \cap D(\sigma')$ e $\sigma(\mathbf{n},\mathbf{p}) = \sigma'(\mathbf{n},\mathbf{p})$.

3.7.15 Definição: Se **I** e **I'** são **LQC**-interpretações para uma coleção de fórmulas Γ, dizemos que **I** e **I'** coincidem em Γ se, para todo $\mathbf{P} \in \Gamma$, **I** e **I'** coincidem em **P**.

3.7.16 Lema: (**Coincidência entre LQC-interpretações**)
Se **I** e **I'** são **LQC**-interpretações, então seguintes proposições são válidas:

- Se **I** e **I'** coincidem em **t**, então $\mathbf{I}_D(\mathbf{t}) = \mathbf{I'}_D(\mathbf{t})$;
- Se **I** e **I'** coincidem em **P**, então $\mathbf{I}_V(\mathbf{P}) = \mathbf{I'}_V(\mathbf{P})$.

- Se **I** e **I**’ coincidem em Γ, então **I** satisfaz Γ se, e somente se, **I**’ satisfaz Γ.

Desenvolvemos aqui as principais idéias da semântica clássica nos níveis proposicional e quantificacional, as quais são um dos pilares conceituais para a particularização do método dos tablôs desenvolvido no próximo capítulo.

Capítulo 4

O Método dos Tablôs

4.1 Introdução

Um *tablô* é uma árvore de fórmulas em uma dada lógica. A partir de uma fórmula inicial, a qual está em geral relacionada com uma possível tese de uma dada lógica ou com a sua negação, é gerada uma árvore ou tablô inicial.

Esta árvore é a primeira de uma seqüência de árvores, onde cada árvore não inicial sucede a anterior através da aplicação de uma regra de expansão a um nó não usado ou marcado. Cada ramo desta árvore em evolução pode estar aberto ou fechado, segundo um critério de fechamento previamente dado. Ramos fechados não crescem mais, enquanto que os abertos são expandidos através da aplicação de regras aos seus nós ainda não usados. Este processo pára quando for encontrado um tablô com todos os ramos fechados, ou quando for encontrado um ramo aberto em que todos os nós não usados não forem aplicáveis a nenhuma regra. No primeiro caso, se for constatada a correção do algoritmo com respeito à lógica considerada, então a possível tese é válida, enquanto que, no segundo caso, se for confirmada a completude do algoritmo, então a possível tese não é válida.

Desenvolveremos aqui um *método dos tablôs por prova direta*, em contraste ao *método tradicional*. O método dos tablôs por refutação, cujos principais precursores foram Beth [5] e Hintikka [31], é um processo algorítmico capaz de provar um dado teorema pelo insucesso de uma busca exaustiva de uma interpretação que satisfaça a sua negação [11]. De forma semelhante, o método dos tablôs por prova direta também é um processo algorítmico, mas sua principal diferença, em relação ao método indireto, está em provar uma dada tese de forma direta, ou seja, sem precisar negá-la.

Iremos nos reportar também ao método dos tablôs por prova direta por *método dos tablôs direto* ou *método direto*.

O *método dos tablôs direto* consiste em desenvolver uma árvore de fórmulas (tablô) a partir de um tablô inicial com um único nó, contendo a possível tese.

4.2 Sistemas de Tablôs

Exporemos aqui as idéias básicas concernentes ao *método dos tablôs* de uma forma abstrata, de modo independente do sistema lógico relacionado.

Consideraremos implicitamente em toda esta seção duas lógicas **L** e **L**', onde **L** e **L**' são respectivamente linguagens para a primeira e para a segunda lógica. A lógica **L** é aquela para a qual se objetiva construir um sistema de tablôs, o qual será definido a seguir, enquanto que a lógica **L**' é auxiliar, isto é, a mesma é definida apenas para facilitar, no caso do método dos tablôs, o estudo da lógica principal. O processo algorítmico envolverá sempre árvores de fórmulas de **L**', isto é, de fórmulas de uma linguagem da lógica auxiliar.

Para um sistema de tablôs as linguagens de trabalho são indispensáveis, pois através delas é possível definir adequadamente os domínios das regras dos sistemas aqui expostos. Em [8], [10] e [11] são citadas duas possíveis razões para o seu uso:

(1) Para o tratamento dos quantificadores, nas lógicas por nós conhecidas, é necessário acrescentar uma infinidade de constantes ao alfabeto da linguagem inicial.

(2) Em algumas ocasiões a linguagem da lógica inicial não é diretamente adequada para uma análise semântica; nestes casos pode ser indispensável o acréscimo de novos sinais à linguagem inicial, rumo à construção de uma lógica de trabalho passível de uma análise semântica, sendo extensão conservativa da primeira.

Uma outra situação não enquadrável nas duas razões que acabamos de citar dá-se no caso em que não é encontrada uma lógica de trabalho adequada como extensão conservativa da primeira. Neste caso busca-se uma lógica de trabalho que, embora não sendo extensão conservativa, seja, a menos de uma tradução, isomorfa em um certo sentido à lógica inicial, dentro das fórmulas da linguagem de trabalho traduzíveis à linguagem inicial.

4.2.1 Notação: Em toda esta seção as letras **L** e **L**' denotam respectivamente *linguagens formais para* lógicas não necessariamente idênticas.

4.2.2 Definição: Um *tablô em* uma linguagem **L**' é uma *árvore* de nós cujo conteúdo é, no mínimo, uma fórmula de **L**' e uma *marca*, a qual, por convenção, pertence ao conjunto $\{0, 1\}$. Se η é um nó, denotamos a fórmula de η por **form(**η**)**. Dizemos que um nó está *marcado* se a sua marca é 1. Dizemos que um nó está em **L**' se sua fórmula é de **L**'.

4.2.3 Definição: Uma árvore finita preenchendo as condições da definição acima é dita um *tablô finito*.

4.2.4 Definição: Um *ramo em* **L**' é um ramo de um tablô em **L**'.

4.2.5 Definição: Sejam ζ a coleção de todos os tablôs finitos em **L**', $\mathcal{F}(\zeta)$ a coleção de todos os subconjuntos finitos de ζ, θ a coleção de todos os ramos em **L**', e θ' a coleção de todos os ramos finitos em **L**'. Uma *função de* *inicialização para* **L** é uma função de **L** em ζ. Um *critério de fechamento em* **L**' é uma função de θ em $\{$**aberto**, **fechado**$\}$. Uma *regra em* **L**' é uma função parcial de **L**' $\times$ θ' em $\mathcal{F}(\zeta)$, cujo domínio é da forma $\mathbf{L}_0 \times \theta$', onde $\mathbf{L}_0 \subseteq$ **L**'.

4.2.6 Definição: Um *sistema de tablôs* **S** = $\langle$**L**, **L**', $\mathcal{I}$, $\mathcal{C}$, $\mathcal{R}\rangle$, é uma quíntupla ordenada **S** = $\langle$**L**, **L**', $\mathcal{I}$, $\mathcal{C}$, $\mathcal{R}\rangle$, onde **L** e **L**' são linguagens formais, $\mathcal{I}$ é uma função de inicialização para **L**, $\mathcal{C}$ é um critério de fechamento em **L**' e $\mathcal{R}$ é uma coleção de regras em **L**'. Dizemos também que **L** é a *linguagem inicial de* **S**, **L**' é *a linguagem de trabalho de* **S**, $\mathcal{I}$ é *a função de inicialização de* **S**, $\mathcal{C}$ é *o critério de fechamento de* **S** e $\mathcal{R}$ é *a coleção de regras de* **S**. Dizemos também que cada elemento de $\mathcal{R}$ é uma *regra de* **S**.

4.2.7 Notação: Consideraremos doravante, em toda esta seção, que **S** é um sistema de tablôs e que **L**, **L**', $\mathcal{I}$, $\mathcal{C}$ e $\mathcal{R}$ são respectivamente a sua linguagem inicial, a sua linguagem de trabalho, a sua função de inicialização, o seu critério de fechamento e a sua coleção de regras.

Em vários sistemas de tablôs, a função de inicialização cria um tablô inicial composto por apenas um nó, cuja fórmula é a possível tese (a menos de uma possível tradução), no caso do método direto, ou a negação da possível tese (a menos de uma possível tradução), no caso do método da refutação.

4.2.8 Definição: Dizemos que **T** é *o tablô inicial para* **P** em **S** se **P** é uma fórmula de **L** e $\mathcal{I}$(**P**) = **T**. **T** é dito ser um *tablô inicial em* **S** se existe uma fórmula **P** de **L** tal que **T** é o tablô inicial para **P** em **S**.

4.2.9 Definição: Dizemos que uma *regra* **r** de **S** é *aplicável a* uma fórmula **P** de **L**' se o domínio de **r** é da forma $\mathbf{L}_0 \times \theta$', onde $\mathbf{P} \in \mathbf{L}_0$. Uma dada *fórmula* de **L**' é dita ser *excluída em* **S** se nenhuma regra de **S** for aplicável à mesma.

Os tablôs são estendidos através das regras, ou seja, o crescimento das árvores de fórmulas dá-se pela aplicação de regras à fórmulas contidas em nós não marcados.

4.2.10 Definição: Se ρ é um ramo em **L**' e $\mathcal{C}(\rho)$ = **fechado**, então ρ é dito um *ramo fechado em* **S**; caso contrário, ρ é dito um *ramo aberto em* **S**.

4.2.11 Definição: Um *tablô em* **L**' é dito ser *fechado em* **S** se todos os seus ramos forem fechados em **S**.

4.2.12 Definição: Um *ramo em* **L**' é dito ser *exaurido em* **S** se todos os seus nós que possuem fórmulas não excluídas em **S** estiverem marcados.

4.2.13 Notação: A letra **T** seguida ou não de plicas ou subíndices denota um tablô em **L**'.

4.2.14 Definição: **T**' é dito ser uma *extensão imediata de* **T** *em* **S** se existe um nó η em **T** não marcado e uma regra **r** em **S** aplicável a **form**(η) tal que **T**' pode ser obtido de **T** marcando η e acrescentando **r**(**form**(η),ρ) em cada ramo aberto ρ de **T** em que η figure.

4.2.15 Definição: Dizemos que $(\mathbf{T}_i)_{i\in I}$ é *uma seqüência de desenvolvimento de tablôs em* **S** se as seguintes condições forem satisfeitas:
- I é um **k**-segmento, para algum $\mathbf{k} \in \mathbb{N}$ ou I = $\mathbb{N}$.
- Para cada $\mathbf{n} \in$ I tal que $\mathbf{n}+1 \in$ I, $\mathbf{T}_{n+1}$ é uma extensão imediata de $\mathbf{T}_n$ em **S**.

4.2.16 Definição: Dizemos que $(\mathbf{T}_i)_{i\in I}$ é *uma seqüência de desenvolvimento de tablôs para* **P** *em* **S** se as seguintes condições forem satisfeitas:
- **P** é uma fórmula em **L**.
- $(\mathbf{T}_i)_{i\in I}$ é uma seqüência de desenvolvimento de tablôs em **S**.
- $\mathbf{T}_0$ é o tablô inicial para **P** em **S**.

4.2.17 Notação: $(\mathbf{T}_i)_{i\in I}$ e $(\mathbf{T}_i)_{i\in \mathbb{N}}$ são seqüências de desenvolvimento de tablôs em **S**.

4.2.18 Definição: Dizemos que **T**' é *um desenvolvimento de* **T** *em* **S** se existir $(\mathbf{T}_i)_{i\in I}$ e existirem **n**,**n**' $\in$ I tal que $\mathbf{n} \leq \mathbf{n}'$, **T** = $\mathbf{T}_n$ e **T**' = $\mathbf{T}_{n'}$.

4.2.19 Definição: $(\mathbf{T}_i)_{i\in I}$ é dita ser (*uma seqüência de desenvolvimento de tablôs em* **S**) *completa* se as seguintes cláusulas forem satisfeitas:
- Se I é um **k**-segmento para algum $\mathbf{k} \in \mathbb{N}$, então $(\mathbf{T}_i)_{i\in I}$ termina com um tablô fechado em **S** ou com um tablô possuindo pelo menos um ramo exaurido aberto em **S**.

- Se I = $\mathbb{N}$, então $(\mathbf{T}_i)_{i \in I}$ tende para uma *árvore limite* possuindo um ramo infinito exaurido aberto.

4.2.20 Definição: Dizemos que **T** é *um tablô para* **P** *em* **S** se **P** é uma fórmula de **L** e **T** é *um desenvolvimento do tablô inicial para* **P** *em* **S**.

4.2.21 Teorema: Toda fórmula em **L** admite uma seqüência de desenvolvimento de tablôs em **S** completa em **S**.

4.3 Provas Gerais de Correção e Completude de um Sistema de Tablôs com respeito a uma Lógica Arbitrária

É possível verificar na literatura existente diversas provas de correção e completude para o método dos tablôs, como por exemplo em [44] e [3]. Também em [12], [9] e [7] são demonstradas as referidas provas para este método para as lógicas paraconsistentes C1 e C1*, definidas por Newton da Costa. Encontramos também em [11] provas de correção e completude de sistemas de tablôs para os cálculos paraconsistentes e/ou paracompletos C1, P1 e N1, definidos pelo mesmo autor. Entretanto, todas as fontes por nós conhecidas referem-se ao método indireto, onde o tablô inicial provém da negação da possível conclusão.

Aqui desenvolveremos uma prova de correção e completude para sistemas de tablôs de forma abstrata, isto é, de forma independente da lógica considerada. Este processo servirá como base para os dois próximos capítulos, nos quais serão particularizadas as provas de correção e completude do método dos tablôs por prova direta para a lógica clássica, nos níveis proposicional e quantificacional.

Para fins de simplicidade, consideraremos aqui que as linguagens inicial e de trabalho de um dado sistema de tablôs são linguagens para uma mesma lógica. No caso geral isto não acontece necessariamente, mas, neste trabalho, isto sempre ocorrerá. Em [44], por exemplo, em sua apresentação de sistemas de tablôs para a lógica clássica, as linguagens de trabalho estão vinculadas, implicitamente, a uma lógica distinta da lógica clássica, embora aquela seja estritamente relacionada com esta.

Em vários sistemas de tablôs, como por exemplo nos sistemas definidos nesta dissertação, as linguagens inicial e de trabalho são iguais no nível proposicional, enquanto que, no nível quantificacional, a linguagem de trabalho é obtida da linguagem inicial acrescentando uma infinidade de novas constantes.

4.3.1 Definição: Dizemos que um sistema de tablôs **S** é *correto* com respeito a uma dada lógica **L** dotada de uma semântica de valorações se a existência de um tablô fechado neste sistema para uma dada fórmula **P** em **L** implicar na validade de **P** nesta lógica.

4.3.2 Definição: Dizemos que um sistema de tablôs **S** é *completo* com respeito a uma lógica **L** dotada de uma semântica de valorações se a validade de uma dada fórmula **P** nesta lógica implicar na existência de um tablô fechado para **P** em **S**.

Os *teoremas gerais de correção* e *completude* que serão vistos e desenvolvidos nesta seção servirão de base para provar que um dado sistema de tablôs é correto e completo com respeito a uma dada lógica se certas condições forem atendidas. Veremos nos capítulos seguintes que, em particular, os sistemas de tablôs diretos para a lógica clássica nos níveis proposicional e quantificacional atendem estas condições.

4.3.3 Notação:

- **L** é uma lógica dotada de uma semântica de valorações.
- **L** e **L**' são linguagens para **L**.
- **S** é um sistema de tablôs, onde **L** e **L**' são, respectivamente, as linguagens inicial e de trabalho.

4.3.4 Definição: Um *ramo em* **L**' é ***L**-válido* se a coleção de suas fórmulas for **L**-válida.

4.3.5 Definição: Um *tablô em* **L**' é ***L****-válido* se, e somente se, todos os seus ramos forem **L**-válidos.

4.3.6 Definição: **S** *preserva a* ***L****-validade na retração* se, dada uma seqüência $(\mathbf{T}_i)_{i\in I}$ de desenvolvimento de tablôs em **S**, para cada **n** > 0 tal que **n** $\in$ I, se $\mathbf{T}_n$ é **L**-válido, então $\mathbf{T}_{n-1}$ é **L**-válido.

As *condições de correção* de um dado sistema de tablôs com respeito a uma dada lógica são cumpridas pela validade de uma fórmula em decorrência da validade de seu tablô inicial, pela preservação retroativa da validade de um tablô em qualquer seqüência de desenvolvimento, e pela validade de um tablô fechado.

4.3.7 Teorema: (**das Condições de Correção**)

Se
- $*$ a **L**-validade de um tablô inicial para **P** em **S** implica na **L**-validade de **P**,
- $*$ **S** preserva a **L**-validade na retração,
- $*$ todo tablô fechado em **S** é **L**-válido,

então a existência de um tablô fechado para **P** em **S** implica que **P** é **L**-válido.

Prova:
Dado um tablô fechado **T** para **P** em **S**, existe uma seqüência $(\mathbf{T}_i)_{i\in I}$ de desenvolvimento de tablôs em **S** tal que $\mathbf{T}_0$ é o tablô inicial para **P** em **S**, e $\mathbf{T}_n$ = **T**, onde **n** é o elemento máximo de I.
Pela terceira condição de correção, temos que **T** = $\mathbf{T}_n$ é **L**-válido, e daí, pela segunda condição de correção, temos que $\mathbf{T}_0$ é **L**-válido, e daí, pela primeira condição de correção, temos que **P** é **L**-válido. **c.q.d.**

As *condições de completude de* um dado sistema de tablôs com respeito a uma dada lógica são cumpridas pela invalidade de um ramo exaurido aberto e pela validade de um tablô inicial em decorrência da validade de sua fórmula.

4.3.8 Teorema: (das Condições de Completude)

Se $\begin{cases} * \text{ todo ramo exaurido aberto em } \mathbf{S} \text{ é } \mathsf{L}\text{-inválido,} \\ * \text{ a } \mathsf{L}\text{-validade de } \mathbf{P} \text{ implica na } \mathsf{L}\text{-validade do tablô inicial} \\ \quad \text{para } \mathbf{P} \text{ em } \mathbf{S}, \end{cases}$

então, se **P** é **L**-válido, toda seqüência de desenvolvimento completa de tablôs para **P** em **S** termina com um tablô fechado para **P** em **S**.

Prova:
Suponhamos que **P** é **L**-válido.
Pelo teorema 4.2.21, temos que existe uma seqüência $(\mathbf{T}_i)_{i\in I}$ de desenvolvimento de tablôs em **S** completa para **P**.
Se $(\mathbf{T}_i)_{i\in I}$ não termina com um tablô fechado em **S**, então duas situações podem ocorrer:
(1) $(\mathbf{T}_i)_{i\in I}$ termina com um tablô possuindo um ramo exaurido aberto;
(2) I = $\mathbb{N}$ e $(\mathbf{T}_i)_{i\in \mathbb{N}}$ tende para uma árvore limite possuindo um ramo exaurido aberto em **S**.
Em ambos os casos temos que o tablô inicial para **P** em **S**, $\mathbf{T}_0$, está contido em uma árvore de fórmulas possuindo um ramo exaurido aberto ρ em **S**.
Seja ρ' o ramo de T_0 incluído em ρ.
Pela primeira condição de completude, temos que ρ é **L**-inválido, donde ρ' também é **L**-inválido, daí T_0 é **L**-inválido, e portanto, pela segunda condição de completude, **P** é **L**-inválido, o que contradiz a suposição inicial.
Portanto $(\mathbf{T}_i)_{i\in I}$ termina com um tablô fechado em **S**. **c.q.d.**

4.3.9 Corolário: (das Condições de Correção e Completude)
Se **S** satisfaz as condições de correção e completude dadas nas formulações dos teoremas 4.3.7 e 4.3.8, então as seguintes proposições são equivalentes:
- **P** é **L**-válido;
- toda seqüência de desenvolvimento completa de tablôs para **P** em **S** termina com um tablô fechado para **P** em **S**;
- existe um tablô fechado para **P** em **S**.

Os dois próximos capítulos são uma aplicação do conteúdo desenvolvido neste para as lógicas proposicional e quantificacional clássica.

Capítulo 5

Um Sistema de Tablôs por Prova Direta para a Lógica Proposicional Clássica

5.1 Definição do Sistema

Definiremos aqui um *sistema de tablôs por prova direta*, notado por **STD**, para a lógica clássica no nível proposicional. A sigla "**STD**" abrevia a expressão "sistema de tablôs direto".

5.1.1 Notação: **L** é uma linguagem para **LPC**.

5.1.2 Definição: A *linguagem inicial* e a de *trabalho* de **STD** são iguais, e as referenciaremos, nesta seção, por **L**.

5.1.3 Definição: *A função de inicialização de* **STD** associa uma fórmula **P** de **L** a um tablô com um único nó não marcado, cuja fórmula é **P**.

5.1.4 Definição: Um *ramo é fechado em* **STD** se ele possuir duas fórmulas contraditórias.

5.1.5 Notação: Usaremos aqui, graficamente, o sinal de *ticagem* "✓" significando o valor 1 da marca, e sua ausência como o valor 0 da mesma.

A coleção de regras de **STD** possui nove regras. O crescimento de um tablô é dado pela aplicação sucessiva das regras a nós ainda não *tica-*

dos. A expansão se dá em todos os ramos abertos abaixo do nó utilizado. As regras aqui desenvolvidas têm a característica de gerar até dois novos ramos.

As regras definidas abaixo devem ser aplicadas até que seja atingido um tablô fechado ou um tablô exaurido aberto em **STD**, através de uma seqüência de desenvolvimento completa em **STD**, a partir de um tablô inicial.

Descreveremos a seguir cada regra de **STD**. As definições das regras deste sistema estão relacionadas com os conectivos lógicos das fórmulas para as quais elas são aplicadas.

Todas as regras de **STD** serão definidas graficamente, dando as fórmulas dos nós que compõem os tablôs resultantes.

Implicação (**P** → **Q**): Esta regra associa a cada ramo contendo **P** → **Q** e à fórmula **P**→**Q** um único tablô com um único ramo contendo dois nós não marcados, cujas fórmulas são respectivamente ¬**P** e **Q**.

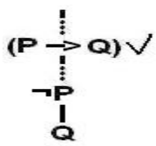


Figura 5.1: Implicação

Conjunção (**P** ∧ **Q**): Esta regra associa a cada ramo contendo **P** ∧ **Q** e à fórmula **P** ∧ **Q** dois tablôs, cada um deles possuindo um único ramo contendo um único nó, cujas fórmulas são respectivamente **P** e **Q**.

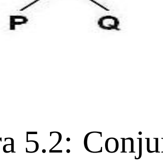


Figura 5.2: Conjunção

Disjunção (**P** ∨ **Q**): Esta regra associa a cada ramo contendo **P** ∨ **Q** e à fórmula **P** ∨ **Q** um único tablô com um único ramo contendo dois nós não marcados, cujas fórmulas são respectivamente **P** e **Q**.

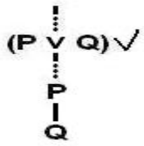


Figura 5.3: Disjunção

Negação de Implicação (¬(**P** → **Q**)): Esta regra associa a cada ramo contendo ¬(**P** → **Q**) e à fórmula ¬(**P** → **Q**) dois tablôs, cada um deles possuindo um único ramo contendo um único nó não marcado, em que a fórmula do primeiro é **P** e a do segundo é ¬**Q**.

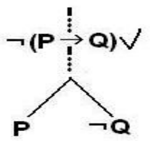


Figura 5.4: Negação da Implicação

Negação de Conjunção (¬(**P** ∧ **Q**)): Esta regra associa a cada ramo contendo ¬(**P** ∧ **Q**) e à fórmula ¬(**P** ∧ **Q**) um único tablô com um único ramo contendo dois nós não marcados, cujas fórmulas são respectivamente ¬**P** e ¬**Q**.

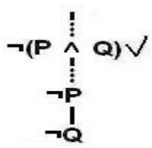


Figura 5.5: Negação de Conjunção

Negação de Disjunção (¬(**P** ∨ **Q**)): Esta regra associa a cada ramo contendo ¬(**P** ∨ **Q**) e à fórmula ¬(**P** ∨ **Q**) dois tablôs, cada um deles possuindo um único ramo contendo um único nó não marcado, em que a fórmula do primeiro é ¬**P** e a do segundo é ¬**Q**.

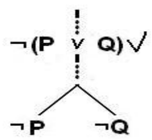


Figura 5.6: Negação de Disjunção

Negação de Negação (¬¬**P**): Esta regra associa a cada ramo contendo ¬¬**P** e à fórmula ¬¬**P** um único tablô com um único nó não marcado cuja fórmula é **P**.

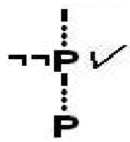


Figura 5.7: Negação de Negação

Equivalência (**P** ↔ **Q**): Esta regra associa a cada ramo contendo **P** ↔ **Q** e à fórmula **P** ↔ **Q** dois tablôs, cada um deles possuindo um único ramo contendo dois nós não marcados, o primeiro contendo as fórmulas ¬**P** e **Q**, e o segundo contendo as fórmulas **P** e ¬**Q**.

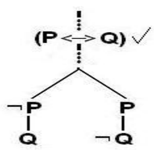


Figura 5.8: Equivalência

Negação de Equivalência (¬(**P** ↔ **Q**)): Esta regra associa a cada ramo contendo ¬(**P** ↔ **Q**) e à fórmula ¬(**P** ↔ **Q**) dois tablôs, cada um deles possuindo um único ramo contendo dois nós não marcados, o primeiro contendo as fórmulas **P** e **Q**, e o segundo contendo as fórmulas ¬**P** e ¬**Q**.

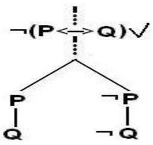


Figura 5.9: Negação de Equivalência

5.2 Prova da Correção de STD

Um dado sistema de tablôs é correto em relação a **LPC** se ele não é capaz de provar algo que não seja válido em **LPC**.

Mostraremos a seguir a correção de **STD** com respeito a **LPC**, com base nas condições gerais de correção vistas no capítulo anterior.

5.2.1 Lema: Se o tablô inicial para **P** em **STD** é **LPC**-válido, então **P** é **LPC**-válido.
Prova:
Seja $\mathbf{T}_0$ o tablô inicial para **P** em **STD**.
Por definição, $\mathbf{T}_0$ possui um único nó cujo conteúdo é **P**.
Se $\mathbf{T}_0$ é **LPC**-válido em **STD**, então todos os ramos de $\mathbf{T}_0$ são **LPC**-válidos, e daí o seu único ramo cuja coleção de fórmulas é $\{\mathbf{P}\}$ é **LPC**-válido, donde $\{\mathbf{P}\}$ é **LPC**-válido, e portanto **P** é **LPC**-válido. **c.q.d.**

5.2.2 Terminologia: Para simplificar denominaremos nesta seção uma **LPC**-valoração por valoração.

5.2.3 Lema: As seguintes proposições são verdadeiras:

(i) Se $\mathbf{P} \rightarrow \mathbf{Q} \in \Gamma$ e $\Gamma \cup \{\neg\mathbf{P}, \mathbf{Q}\}$ é **LPC**-válido, então Γ é **LPC**-válido.

(ii) Se $\mathbf{P} \wedge \mathbf{Q} \in \Gamma$, $\Gamma \cup \{\mathbf{P}\}$ e $\Gamma \cup \{\mathbf{Q}\}$ são **LPC**-válidos, então Γ é **LPC**-válido.

(iii) Se $\mathbf{P} \vee \mathbf{Q} \in \Gamma$ e $\Gamma \cup \{\mathbf{P},\mathbf{Q}\}$ é **LPC**-válido, então Γ é **LPC**-válido.

(iv) Se $\neg(\mathbf{P} \rightarrow \mathbf{Q}) \in \Gamma$, $\Gamma \cup \{\mathbf{P}\}$ e $\Gamma \cup \{\neg\mathbf{Q}\}$ são **LPC**-válidos, então Γ é **LPC**-válido.

(v) Se $\neg(\mathbf{P} \wedge \mathbf{Q}) \in \Gamma$ e $\Gamma \cup \{\neg\mathbf{P}, \neg\mathbf{Q}\}$ é **LPC**-válido, então Γ é **LPC**-válido.

(vi) Se $\neg(\mathbf{P} \vee \mathbf{Q}) \in \Gamma$, $\Gamma \cup \{\neg\mathbf{P}\}$ e $\Gamma \cup \{\neg\mathbf{Q}\}$ são **LPC**-válidos, então Γ é **LPC**-válido.

(vii) Se $\neg\neg\mathbf{P} \in \Gamma$ e $\Gamma \cup \{\mathbf{P}\}$ é **LPC**-válido, então Γ é **LPC**-válido.

(viii) Se $\mathbf{P} \leftrightarrow \mathbf{Q} \in \Gamma$, $\Gamma \cup \{\mathbf{P},\neg\mathbf{Q}\}$ e $\Gamma \cup \{\neg\mathbf{P},\mathbf{Q}\}$ são **LPC**-válidos, então Γ é **LPC**-válido.

(ix) Se $\neg(\mathbf{P} \leftrightarrow \mathbf{Q}) \in \Gamma$, $\Gamma \cup \{\mathbf{P}, \mathbf{Q}\}$ e $\Gamma \cup \{\neg\mathbf{P}, \neg\mathbf{Q}\}$ são **LPC**-válidos, então Γ é **LPC**-válido.

Prova de (i) :

Suponha que "$\mathbf{P} \rightarrow \mathbf{Q}$" $\in \Gamma$ e $\Gamma \cup \{\neg\mathbf{P}, \mathbf{Q}\}$ é **LPC**-válido.

Dada uma valoração **V**, temos que **V** satisfaz pelo menos uma fórmula **R** de $\Gamma \cup \{\neg\mathbf{P}, \mathbf{Q}\}$.

Se $\mathbf{R} \in \Gamma$, então **V** obviamente satisfaz pelo menos uma fórmula de Γ.

Se **R** é $\neg\mathbf{P}$, então $\mathbf{V}(\neg\mathbf{P}) = \mathbf{v}$, donde $\mathbf{V}(\mathbf{P} \rightarrow \mathbf{Q}) = \mathbf{v}$, e daí **V** satisfaz a fórmula $\mathbf{P} \rightarrow \mathbf{Q}$ de Γ.

Se **R** é **Q**, então $\mathbf{V}(\mathbf{Q}) = \mathbf{v}$, donde $\mathbf{V}(\mathbf{P} \rightarrow \mathbf{Q}) = \mathbf{v}$, e daí, **V** satisfaz a fórmula $\mathbf{P} \rightarrow \mathbf{Q}$ de Γ.

Portanto Γ é **LPC**-válido.

Prova de (ii) :

Suponha que "$\mathbf{P} \wedge \mathbf{Q}$" $\in \Gamma$, $\Gamma \cup \{\mathbf{P}\}$ e $\Gamma \cup \{\mathbf{Q}\}$ são **LPC**-válidos.

Dada uma valoração **V**, temos que **V** satisfaz pelo menos uma fórmula **R** de $\Gamma \cup \{\mathbf{P}\}$ e pelo menos uma fórmula **S** de $\Gamma \cup \{\mathbf{Q}\}$.

Se $\mathbf{R} \in \Gamma$ ou $\mathbf{S} \in \Gamma$, temos trivialmente que **V** satisfaz pelo menos uma fórmula de Γ, daí podemos nos ater ao caso em que $\mathbf{R} = \mathbf{P}$ e $\mathbf{S} = \mathbf{Q}$, donde $\mathbf{V}(\mathbf{P}) = \mathbf{V}(\mathbf{Q}) = \mathbf{v}$, daí $\mathbf{V}(\mathbf{P} \wedge \mathbf{Q}) = \mathbf{v}$, e daí, como $\mathbf{P} \wedge \mathbf{Q} \in \Gamma$, temos novamente que **V** satisfaz pelo menos uma fórmula de Γ.

Portanto Γ é **LPC**-válido.

Prova de (iii) :

Suponha que "$\mathbf{P} \vee \mathbf{Q}$" $\in \Gamma$ e $\Gamma \cup \{\mathbf{P},\mathbf{Q}\}$ é **LPC**-válido.

Dada uma valoração **V**, temos que **V** satisfaz pelo menos uma fórmula **R** de $\Gamma \cup \{\mathbf{P}, \mathbf{Q}\}$.

Se $\mathbf{R} \in \Gamma$, então **V** obviamente satisfaz pelo menos uma fórmula de Γ.

Se **R** é **P**, então $\mathbf{V}(\mathbf{P}) = \mathbf{v}$, donde $\mathbf{V}(\mathbf{P} \vee \mathbf{Q}) = \mathbf{v}$, daí **V** satisfaz a fórmula $\mathbf{P} \vee \mathbf{Q}$ de Γ.

Se **R** é **Q**, então $\mathbf{V}(\mathbf{Q}) = \mathbf{v}$, donde $\mathbf{V}(\mathbf{P} \vee \mathbf{Q}) = \mathbf{v}$, daí **V** satisfaz a fórmula $\mathbf{P} \vee \mathbf{Q}$ de Γ.

Portanto Γ é **LPC**-válido.

Prova de iv :

Suponha que "$\neg(\mathbf{P} \rightarrow \mathbf{Q})$" $\in \Gamma$, $\Gamma \cup \{\mathbf{P}\}$ e $\Gamma \cup \{\neg\mathbf{Q}\}$ são **LPC**-válidos.

Dada uma valoração **V**, temos que **V** satisfaz pelo menos uma fórmula **R**

de $\Gamma \cup \{\mathbf{P}\}$ e pelo menos uma fórmula **S** de $\Gamma \cup \{\neg\mathbf{Q}\}$.
Se $\mathbf{R} \in \Gamma$ ou $\mathbf{S} \in \Gamma$, temos obviamente que **V** satisfaz pelo menos uma fórmula de Γ, daí podemos nos ater ao caso em que $\mathbf{R} = \mathbf{P}$ e $\mathbf{S} = \neg\mathbf{Q}$, donde $\mathbf{V}(\mathbf{P}) = \mathbf{V}(\neg\mathbf{Q}) = \mathbf{v}$, daí $\mathbf{V}(\neg(\mathbf{P} \to \mathbf{Q})) = \mathbf{v}$, e daí, como $\neg(\mathbf{P} \to \mathbf{Q}) \in \Gamma$, temos novamente que **V** satisfaz pelo menos uma fórmula de Γ.
Portanto Γ é **LPC**-válido.

Prova de (v) :
Suponha que "$\neg(\mathbf{P} \wedge \mathbf{Q})$" $\in \Gamma$ e $\Gamma \cup \{\neg\mathbf{P}, \neg\mathbf{Q}\}$ é **LPC**-válido.
Dada uma valoração **V**, temos que **V** satisfaz pelo menos uma fórmula **R** de $\Gamma \cup \{\neg\mathbf{P}, \neg\mathbf{Q}\}$.
Se $\mathbf{R} \in \Gamma$, então **V** obviamente satisfaz pelo menos uma fórmula de Γ.
Se **R** é $\neg\mathbf{P}$, então $\mathbf{V}(\neg\mathbf{P}) = \mathbf{v}$, donde $\mathbf{V}(\neg(\mathbf{P} \wedge \mathbf{Q})) = \mathbf{v}$, daí **V** satisfaz a fórmula $\neg(\mathbf{P} \wedge \mathbf{Q})$ de Γ.
Se **R** é $\neg\mathbf{Q}$, então $\mathbf{V}(\neg\mathbf{P}) = \mathbf{v}$, donde $\mathbf{V}(\neg(\mathbf{P} \wedge \mathbf{Q})) = \mathbf{v}$, daí **V** satisfaz a fórmula $\neg(\mathbf{P} \wedge \mathbf{Q})$ de Γ.
Portanto Γ é **LPC**-válido.

Prova de (vi) :
Suponha que "$\neg(\mathbf{P} \vee \mathbf{Q})$" $\in \Gamma$, $\Gamma \cup \{\neg\mathbf{P}\}$ e $\Gamma \cup \{\neg\mathbf{Q}\}$ são **LPC**-válidos.
Dada uma valoração **V**, temos que **V** satisfaz pelo menos uma fórmula **R** de $\Gamma \cup \{\neg\mathbf{P}\}$ e pelo menos uma fórmula **S** de $\Gamma \cup \{\neg\mathbf{Q}\}$.
Se $\mathbf{R} \in \Gamma$ ou $\mathbf{S} \in \Gamma$, temos obviamente que **V** satisfaz pelo menos uma fórmula de Γ, daí podemos nos ater ao caso em que $\mathbf{R} = \neg\mathbf{P}$ e $\mathbf{S} = \neg\mathbf{Q}$, donde $\mathbf{V}(\neg\mathbf{P}) = \mathbf{V}(\neg\mathbf{Q}) = \mathbf{v}$, daí $\mathbf{V}(\neg(\mathbf{P} \vee \mathbf{Q})) = \mathbf{v}$, e daí, como $\neg(\mathbf{P} \vee \mathbf{Q}) \in \Gamma$, temos novamente que **V** satisfaz pelo menos uma fórmula de Γ.
Portanto Γ é **LPC**-válido.

Prova de (vii) :
Suponha que "$\neg\neg\mathbf{P}$" $\in \Gamma$ e $\Gamma \cup \{\mathbf{P}\}$ é **LPC**-válido.
Dada uma valoração **V**, temos que **V** satisfaz pelo menos uma fórmula **R** de $\Gamma \cup \{\mathbf{P}\}$.
Se $\mathbf{R} \in \Gamma$, então **V** satisfaz pelo menos uma fórmula de Γ.
Se $\mathbf{R} = \mathbf{P}$, então $V(\mathbf{P}) = \mathbf{v}$, donde $\mathbf{V}(\neg\neg\mathbf{P}) = \mathbf{v}$, daí **V** satisfaz a fórmula $\neg\neg\mathbf{P}$ de Γ.
Portanto Γ é **LPC**-válido.

Prova de (viii) :
Suponha que "$\mathbf{P} \leftrightarrow \mathbf{Q}$" $\in \Gamma$, $\Gamma \cup \{\mathbf{P}, \neg\mathbf{Q}\}$ e $\Gamma \cup \{\neg\mathbf{P}, \mathbf{Q}\}$ são **LPC**-válidos.
Dada uma valoração **V**, temos que **V** satisfaz pelo menos uma fórmula **R** de $\Gamma \cup \{\mathbf{P}, \neg\mathbf{Q}\}$ e pelo menos uma fórmula **S** de $\Gamma \cup \{\neg\mathbf{P}, \mathbf{Q}\}$.

Se **R** $\in \Gamma$ ou **S** $\in \Gamma$, temos trivialmente que **V** satisfaz pelo menos uma fórmula de Γ.
Se **R** é **P** e **S** é **Q**, então **V**(**P**) = **V**(**Q**) = **v**, donde **V**(**P** $\leftrightarrow$ **Q**) = **v**.
Se **R** é $\neg$**P** e **S** é $\neg$**Q**, então **V**($\neg$**P**) = **V**($\neg$**Q**) = **v**, donde **V**(**P**) = **f** e **V**(**Q**) = **f**, e daí **V**(**P** $\leftrightarrow$ **Q**) = **v**.
Logo, nos dois últimos casos possíveis, **V** satisfaz pelo menos uma fórmula de Γ.
Portanto Γ é **LPC**-válido.

Prova de (ix) :
Suponha que "$\neg$(**P** $\leftrightarrow$ **Q**)" $\in \Gamma$, $\Gamma \cup \{$**P**, **Q**$\}$ e $\Gamma \cup \{\neg$**P**, $\neg$**Q**$\}$ são **LPC**-válidos.
Dada uma valoração **V**, temos que **V** satisfaz pelo menos uma fórmula **R** de $\Gamma \cup \{$**P**, **Q**$\}$ e pelo menos uma fórmula **S** de $\Gamma \cup \{\neg$**P**, $\neg$**Q**$\}$.
Se **R** $\in \Gamma$ ou **S** $\in \Gamma$, temos trivialmente que **V** satisfaz pelo menos uma fórmula de Γ.
Se **R** é **P** e **S** é $\neg$**Q**, então **V**(**P**) = **v** e **V**($\neg$**Q**) = **v**, donde **V**(**Q**) = **f**, e daí **V**($\neg$(**P** $\leftrightarrow$ **Q**)) = **v**.
Se $\neg$**R** é **P** e **S** é **Q**, então **V**($\neg$**P**) = **v** e **V**(**Q**) = **v**, donde **V**(**P**) = **f**, e daí **V**($\neg$(**P** $\leftrightarrow$ **Q**)) = **v**.
Logo, nos dois últimos casos possíveis, **V** satisfaz pelo menos uma fórmula de Γ.
Portanto Γ é **LPC**-válido. **c.q.d.**

5.2.4 Lema: Dada uma seqüência $(\mathbf{T}_n)_{n\in I}$ de desenvolvimento de tablôs em **STD**, para cada **n** $\in$ I, se **n** $>$ 0 e $\mathbf{T}_n$ é **LPC**-válido, então $\mathbf{T}_{n-1}$ é **LPC**-válido.
Prova:
Suponha que **n** $\in$ I, **n** > 0, e que $\mathbf{T}_n$ é **LPC**-válido.
Seja ρ um ramo arbitrário de $\mathbf{T}_{n-1}$, e seja Γ a sua coleção de fórmulas.
Se ρ é fechado em **STD**, temos que Γ possui duas fórmulas contraditórias, e daí temos trivialmente que Γ é **LPC**-válido, donde ρ é **LPC**-válido. Daí iremos nos ater doravante ao caso em que ρ é aberto em **STD**.
Seja η o nó de $\mathbf{T}_{n-1}$ que foi usado para $\mathbf{T}_{n-1}$ expandir-se para $\mathbf{T}_n$.
Se $\eta \notin \rho$, temos que ρ também é um ramo de $\mathbf{T}_n$, e daí, como $\mathbf{T}_n$ é **LPC**-válido, temos que ρ é **LPC**-válido, e daí podemos considerar, doravante, que η é um nó de ρ.

Caso a fórmula de η é $\mathbf{P} \rightarrow \mathbf{Q}$, então $\mathbf{T}_n$ possui um ramo estendendo ρ cuja coleção de fórmulas é $\Gamma \cup \{\neg\mathbf{P}, \mathbf{Q}\}$, e daí, pelo lema 5.2.3, Γ é **LPC**-válido.

Caso a fórmula de η é $\mathbf{P} \wedge \mathbf{Q}$, então $\mathbf{T}_n$ possui dois ramos estendendo ρ cujas coleções de fórmulas são $\Gamma \cup \{\mathbf{P}\}$ e $\Gamma \cup \{\mathbf{Q}\}$, e daí, pelo lema 5.2.3, Γ é **LPC**-válido.

Caso a fórmula de η é $\mathbf{P} \vee \mathbf{Q}$, então $\mathbf{T}_n$ possui um ramo estendendo ρ cuja coleção de fórmulas é $\Gamma \cup \{\mathbf{P}, \mathbf{Q}\}$, e daí, pelo lema 5.2.3, Γ é **LPC**-válido.

Caso a fórmula de η é $\neg(\mathbf{P} \rightarrow \mathbf{Q})$, então $\mathbf{T}_n$ possui dois ramos estendendo ρ cujas coleções de fórmulas são $\Gamma \cup \{\mathbf{P}\}$ e $\Gamma \cup \{\neg\mathbf{Q}\}$, e daí, pelo lema 5.2.3, Γ é **LPC**-válido.

Caso a fórmula de η é $\neg(\mathbf{P} \wedge \mathbf{Q})$, então $\mathbf{T}_n$ possui um ramo estendendo ρ cuja coleção de fórmulas é $\Gamma \cup \{\neg\mathbf{P}, \neg\mathbf{Q}\}$, e daí, pelo lema 5.2.3, Γ é **LPC**-válido.

Caso a fórmula de η é $\neg(\mathbf{P} \vee \mathbf{Q})$, então $\mathbf{T}_n$ possui dois ramos estendendo ρ cujas coleções de fórmulas são $\Gamma \cup \{\neg\mathbf{P}\}$ e $\Gamma \cup \{\neg\mathbf{Q}\}$, e daí, pelo lema 5.2.3, Γ é **LPC**-válido.

Caso a fórmula de η é $\neg\neg\mathbf{P}$, então $\mathbf{T}_n$ possui um único ramo estendendo ρ cuja coleção de fórmulas e $\Gamma \cup \{\mathbf{P}\}$, e daí, pelo lema 5.2.3, Γ é **LPC**-válido.

Caso a fórmula de η é $\mathbf{P} \leftrightarrow \mathbf{Q}$, então $\mathbf{T}_n$ possui dois ramos estendendo ρ cujas coleções de fórmulas são $\Gamma \cup \{\mathbf{P}, \neg\mathbf{Q}\}$ e $\Gamma \cup \{\neg\mathbf{P}, \mathbf{Q}\}$, e daí, pelo lema 5.2.3, Γ é **LPC**-válido.

Caso a fórmula de η é $\neg(\mathbf{P} \leftrightarrow \mathbf{Q})$, então $\mathbf{T}_n$ possui dois ramos estendendo ρ cujas coleções de fórmulas são $\Gamma \cup \{\mathbf{P}, \mathbf{Q}\}$ e $\Gamma \cup \{\neg\mathbf{P}, \neg\mathbf{Q}\}$, e daí, pelo lema 5.2.3, Γ é **LPC**-válido.

Daí, em qualquer caso, Γ é **LPC**-válido, donde ρ é **LPC**-válido.
Portanto $\mathbf{T}_{n-1}$ é **LPC**-válido. **c.q.d.**

5.2.5 Lema: Todo tablô fechado em **STD** é **LPC**-válido.
Prova:
Seja **T** um tablô fechado em **STD**.
Dado um ramo ρ de **T**, temos que a coleção de fórmulas de ρ possui duas fórmulas contraditórias, e daí esta coleção é **LPC**-válida, donde ρ é **LPC**-válido. **c.q.d.**

5.2.6 Teorema: **STD** é correto com respeito a **LPC**.
Prova:
Basta aplicar os lemas 5.2.1, 5.2.4 e 5.2.5. **c.q.d.**

5.2.7 Corolário: Se existe um tablô fechado para **P** em **STD**, então **P** é **LPC**-válido.
Prova:
Basta aplicar o teorema 5.2.6 e a definição de correção de um sistema de tablôs com respeito a uma dada lógica. **c.q.d.**

5.3 Prova da Completude de STD

Provaremos aqui a completude de **STD** com respeito a **LPC**, com base no teorema das condições gerais de completude, visto no capítulo anterior.

5.3.1 Lema: Todo ramo exaurido aberto em **STD** é **LPC**-inválido.
Prova:
Seja Γ a coleção de fórmulas de um ramo exaurido aberto em **STD**.
Seja **V** uma valoração, tal que **V**(**P**) = **f** se, e somente se, **P** $\in \Gamma$, para cada letra sentencial **P**.
Provaremos as seguintes propriedades por indução sobre a fórmula **P**.

- Se **P** $\in \Gamma$, então **V**(**P**) = **f**.
- Se $\neg$**P** $\in \Gamma$, então **V**(**P**) = **v**.

Caso i) P é uma letra sentencial.
Se **P** $\in \Gamma$, então, por construção de **V**, **V**(**P**) = **f**.
Se $\neg$**P** $\in \Gamma$, então **P** $\notin \Gamma$, e daí, também por construção de **V**, **V**(**P**) = **v**, logo **V**($\neg$**P**) = **f**.
Caso ii) P é da forma Q $\rightarrow$ R.
Suponhamos, por hipótese de indução, que estas propriedades valham para

Q e **R**.

Se “$\mathbf{Q} \to \mathbf{R}$” $\in \Gamma$, então $\neg\mathbf{Q} \in \Gamma$ e $\mathbf{R} \in \Gamma$. Pela hipótese de indução, temos que $\mathbf{V}(\mathbf{Q}) = \mathbf{v}$ e $\mathbf{V}(\mathbf{R}) = \mathbf{f}$, logo $\mathbf{V}(\mathbf{Q} \to \mathbf{R}) = \mathbf{f}$.

Se $\neg(\mathbf{Q} \to \mathbf{R}) \in \Gamma$, então $\mathbf{Q} \in \Gamma$ ou $\neg\mathbf{R} \in \Gamma$. Pela hipótese de indução, temos que $\mathbf{V}(\mathbf{Q}) = \mathbf{f}$ ou $\mathbf{V}(\neg\mathbf{R}) = \mathbf{f}$, donde $\mathbf{V}(\neg(\mathbf{Q} \to \mathbf{R})) = \mathbf{f}$.

Caso iii) P é da forma Q $\wedge$ R.

Suponhamos, por hipótese de indução, que estas propriedades valham para **Q** e **R**.

Se “$\mathbf{Q} \wedge \mathbf{R}$” $\in \Gamma$, então $\mathbf{Q} \in \Gamma$ ou $\mathbf{R} \in \Gamma$. Pela hipótese de indução, temos que $\mathbf{V}(\mathbf{Q}) = \mathbf{f}$ ou $\mathbf{V}(\mathbf{R}) = \mathbf{f}$, logo $\mathbf{V}(\mathbf{Q} \wedge \mathbf{R}) = \mathbf{f}$.

Se $\neg(\mathbf{Q} \wedge \mathbf{R}) \in \Gamma$, então $\neg\mathbf{Q} \in \Gamma$ e $\neg\mathbf{R} \in \Gamma$. Pela hipótese de indução, temos que $\mathbf{V}(\neg\mathbf{Q}) = \mathbf{V}(\neg\mathbf{R}) = \mathbf{f}$, donde $\mathbf{V}(\neg(\mathbf{Q} \wedge \mathbf{R})) = \mathbf{f}$.

Caso iv) P é da forma Q $\vee$ R.

Suponhamos, por hipótese de indução, que estas propriedades valham para **Q** e **R**.

Se “$\mathbf{Q} \vee \mathbf{R}$” $\in \Gamma$, então $\mathbf{Q} \in \Gamma$ e $\mathbf{R} \in \Gamma$. Pela hipótese de indução, temos que $V(\mathbf{Q}) = \mathbf{f}$ e $\mathbf{V}(\mathbf{R}) = \mathbf{f}$, logo $\mathbf{V}(\mathbf{Q} \vee \mathbf{R}) = \mathbf{f}$.

Se $\neg(\mathbf{Q} \vee \mathbf{R}) \in \Gamma$, então $\neg\mathbf{Q} \in \Gamma$ ou $\neg\mathbf{R} \in \Gamma$. Pela hipótese de indução, temos que $\mathbf{V}(\mathbf{Q}) = \mathbf{v}$ ou $\mathbf{V}(\mathbf{R}) = \mathbf{v}$, e daí, $\mathbf{V}(\neg\mathbf{Q}) = \mathbf{f}$ ou $\mathbf{V}(\neg\mathbf{R}) = \mathbf{f}$, donde $\mathbf{V}(\neg(\mathbf{Q} \vee \mathbf{R})) = \mathbf{f}$.

Caso v) P é da forma $\neg$(Q $\to$ R).

Suponhamos, por hipótese de indução, que estas propriedades valham para **Q** e **R**.

Se “$\neg(\mathbf{Q} \to \mathbf{R}$”) $\in \Gamma$, então $\mathbf{Q} \in \Gamma$ ou $\neg\mathbf{R} \in \Gamma$. Pela hipótese de indução, temos que $\mathbf{V}(\mathbf{Q}) = \mathbf{V}(\mathbf{R}) = \mathbf{f}$, e daí, $\mathbf{V}(\mathbf{Q}) = \mathbf{f}$ e $\mathbf{V}(\neg\mathbf{R}) = \mathbf{v}$, logo $\mathbf{V}(\neg(\mathbf{Q} \to \mathbf{R})) = \mathbf{f}$.

Se “$\neg\neg(\mathbf{Q} \to \mathbf{R}$”) $\in \Gamma$, então $\neg\mathbf{Q} \in \Gamma$ e $\mathbf{R} \in \Gamma$. Pela hipótese de indução, temos que $\mathbf{V}(\mathbf{Q}) = \mathbf{V}(\mathbf{R}) = \mathbf{f}$, e daí, $\mathbf{V}(\neg\mathbf{Q}) = \mathbf{v}$ e $\mathbf{V}(\mathbf{R}) = \mathbf{f}$, donde $\mathbf{V}(\neg\neg(\mathbf{Q} \to \mathbf{R})) = \mathbf{f}$.

Caso vi) P é da forma $\neg$(Q $\wedge$ R).

Suponhamos, por hipótese de indução, que estas propriedades valham para **Q** e **R**.

Se “$\neg(\mathbf{Q} \wedge \mathbf{R}$”) $\in \Gamma$, então $\neg\mathbf{Q} \in \Gamma$ e $\neg\mathbf{R} \in \Gamma$. Pela hipótese de indução, temos que $\mathbf{V}(\mathbf{Q}) = \mathbf{V}(\mathbf{R}) = \mathbf{v}$, e daí, $\mathbf{V}(\neg\mathbf{Q}) = \mathbf{V}(\neg\mathbf{R}) = \mathbf{f}$, logo $\mathbf{V}(\neg(\mathbf{Q} \wedge \mathbf{R})) = \mathbf{f}$.

Se “$\neg\neg(\mathbf{Q} \wedge \mathbf{R}$”) $\in \Gamma$, então $\mathbf{Q} \in \Gamma$ ou $\mathbf{R} \in \Gamma$. Pela hipótese de indução, temos que $\mathbf{V}(\mathbf{Q}) = \mathbf{V}(\mathbf{R}) = \mathbf{f}$, donde $\mathbf{V}(\neg\neg(\mathbf{Q} \wedge \mathbf{R})) = \mathbf{f}$.

Caso vii) P é da forma $\neg$(Q $\vee$ R).
Suponhamos, por hipótese de indução, que estas propriedades valham para **Q** e **R**.
Se "$\neg$(**Q** $\vee$ **R**") $\in \Gamma$, então $\neg$**Q** $\in \Gamma$ ou $\neg$**R** $\in \Gamma$. Pela hipótese de indução, temos que **V**(**Q**) = **v** e **V**(**R**) = **v**, e daí, **V**($\neg$**Q**) = **V**($\neg$**R**) = **f**, logo **V**($\neg$(**Q** $\vee$ **R**)) = **f**.
Se "$\neg\neg$(**Q** $\vee$ **R**") $\in \Gamma$, então **Q** $\in \Gamma$ e **R** $\in \Gamma$. Pela hipótese de indução, temos que **V**(**Q**) = **V**(**R**) = **f**, donde **V**($\neg\neg$(**Q** $\vee$ **R**)) = **f**.

Caso viii) P é da forma Q $\leftrightarrow$ R.
Suponhamos, por hipótese de indução, que estas propriedades valham para **Q** e **R**.
Se "**Q** $\leftrightarrow$ **R**" $\in \Gamma$, então **Q** $\in \Gamma$ e $\neg$**R** $\in \Gamma$, ou $\neg$**Q** $\in \Gamma$ e **R** $\in \Gamma$. Pela hipótese de indução, temos que **V**(**Q**) = **f** e **V**(**R**) = **v**, ou **V**(**Q**) = **v** e **V**(**R**) = **f**, logo **V**(**Q** $\leftrightarrow$ **R**)) = **f**.
Se $\neg$(**Q** $\leftrightarrow$ **R**) $\in \Gamma$, então **Q** $\in \Gamma$ e **R** $\in \Gamma$, ou $\neg$**Q** $\in \Gamma$ e $\neg$**R** $\in \Gamma$. Pela hipótese de indução, temos que **V**(**Q**) = **V**(**R**)= **f**, ou **V**(**Q**) = **V**(**R**) = **v**, e daí, **V**($\neg$**Q**) = **V**($\neg$**R**) = **f**, donde **V**($\neg$(**Q** $\leftrightarrow$ **R**)) = **f**.

Caso ix) P é da forma $\neg$(Q $\leftrightarrow$ R).
Suponhamos, por hipótese de indução, que estas propriedades valham para **Q** e **R**.
Se "$\neg$(**Q** $\leftrightarrow$ **R**)" $\in \Gamma$, então **Q** $\in \Gamma$ e **R** $\in \Gamma$, ou $\neg$**Q** $\in \Gamma$ e $\neg$**R** $\in \Gamma$. Pela hipótese de indução, temos que **V**(**Q**) = **V**(**R**) = **f**, ou **V**(**Q**) = **V**(**R**) = **f**, e daí, **V**($\neg$**Q**) = **V**($\neg$**R**) = **f**, logo **V**($\neg$(**Q** $\leftrightarrow$ **R**)) = **f**.

Se "$\neg\neg$(**Q** $\leftrightarrow$ **R**)" $\in \Gamma$, então **Q** $\in \Gamma$ e $\neg$**R** $\in \Gamma$, ou $\neg$**Q** $\in \Gamma$ e **R** $\in \Gamma$.
Pela hipótese de indução, temos que **V**(**Q**) = **f** e **V**(**R**) = **v**, ou **V**(**Q**) = **v** e **V**(**R**) = **f**, e daí, **V**(**Q**) = **V**($\neg$**R**) = **f**, ou **V**($\neg$**Q**) = **V**(**R**) = **f**, donde **V**($\neg\neg$(**Q** $\leftrightarrow$ **R**)) = **f**. **c.q.d.**

5.3.2 Lema: Se **P** é **LPC**-válido, então o tablô inicial para **P** em **STD** é **LPC**-válido.
Prova:
Suponha que **P** é **LPC**-válido, e seja $\mathbf{T}_0$ o tablô inicial para **P** em **STD**.
Por definição, $\mathbf{T}_0$ possui um único nó cujo conteúdo é **P**.
Portanto, se **P** é **LPC**-válido, então a coleção de fórmulas do único ramo de $\mathbf{T}_0$, $\{\mathbf{P}\}$, é **LPC**-válida, donde $\mathbf{T}_0$ é **LPC**-válido. **c.q.d.**

5.3.3 Teorema: **STD** é completo com respeito a **LPC**.
Prova:
Basta aplicar os lemas 5.3.1 e 5.3.2. **c.q.d.**

5.3.4 Corolário: Se **P** é **LPC**-válido, então toda seqüência de desenvolvimento completa de tablôs para **P** em **STD** termina com um tablô fechado em **STD**.
Prova:
Basta aplicar o teorema 5.3.3 e a definição de completude de um sistema de tablôs com respeito a uma dada lógica. **c.q.d.**

5.3.5 Corolário: As seguintes proposições são equivalentes:
- **P** é **LPC**-válido;
- toda seqüência de desenvolvimento de tablôs para **P** em **STD** termina com um tablô fechado em **STD**;
- existe um tablô fechado para **P** em **STD**.

5.3.6 Notação: Usaremos aqui, graficamente, o sinal "$\smile$" para indicar quando um dado ramo é fechado em **STD**, e o sinal "$\uparrow$" para dizer que um ramo é aberto em **STD**.

5.4 Exemplos

5.4.1 Exemplo com uma fórmula válida em LPC

Dada a fórmula "$\mathbf{P} \wedge (\mathbf{Q} \vee \mathbf{R}) \leftrightarrow (\mathbf{P} \wedge \mathbf{Q}) \vee (\mathbf{P} \wedge \mathbf{R})$", temos que, conforme a figura 5.10, há um tablô fechado para a mesma em **STD**, e daí, conforme o corolário 5.2.7, ela é uma tese em **LPC**. A figura 5.10 nos mostra a seqüência em que os tablôs foram desenvolvidos, sendo que, para cada natural **n** menor ou igual que a profundidade desta árvore, o seu numeral correspondente foi colocado ao lado da representação de cada nó de $\mathbf{T}_n$ que não é nó de $\mathbf{T}_{n-1}$. Esta seqüência termina com um tablô fechado em **STD**, e daí esta lei da distributividade de **LPC** é válida, ou seja, $\models_{\mathbf{LPC}} \mathbf{P} \wedge (\mathbf{Q} \vee \mathbf{R}) \leftrightarrow (\mathbf{P} \wedge \mathbf{Q}) \vee (\mathbf{P} \wedge \mathbf{R})$.

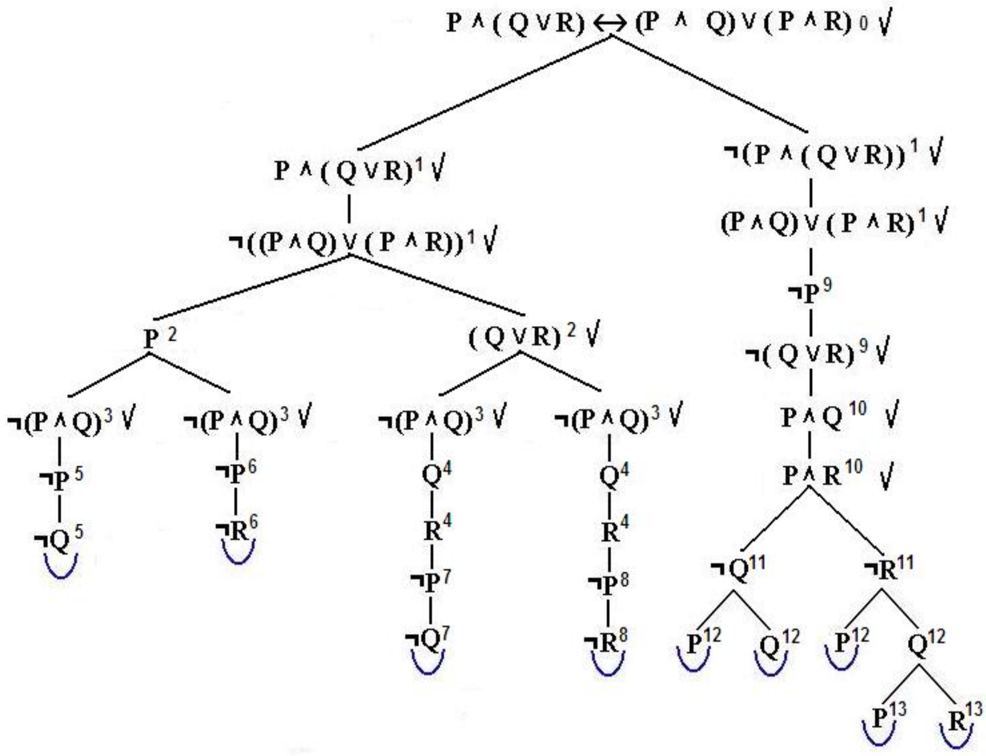


Figura 5.10: Exemplo com uma fórmula válida em **LPC**

5.4.2 Exemplo com uma fórmula inválida em LPC

O tablô para a fórmula "$\mathbf{P} \vee (\mathbf{Q} \wedge \mathbf{R}) \leftrightarrow (\mathbf{P} \vee \mathbf{Q}) \vee (\mathbf{P} \vee \mathbf{R})$", exposto na figura 5.11, possui dois ramos exauridos abertos em **STD**, e daí, com base no corolário 5.3.4, temos que "$\mathbf{P} \vee (\mathbf{Q} \wedge \mathbf{R}) \leftrightarrow (\mathbf{P} \vee \mathbf{Q}) \vee (\mathbf{P} \vee \mathbf{R})$" não é válido em **LPC**, ou seja, $\not\models_{\mathbf{LPC}} \mathbf{P} \vee (\mathbf{Q} \wedge \mathbf{R}) \leftrightarrow (\mathbf{P} \vee \mathbf{Q}) \vee (\mathbf{P} \vee \mathbf{R})$.

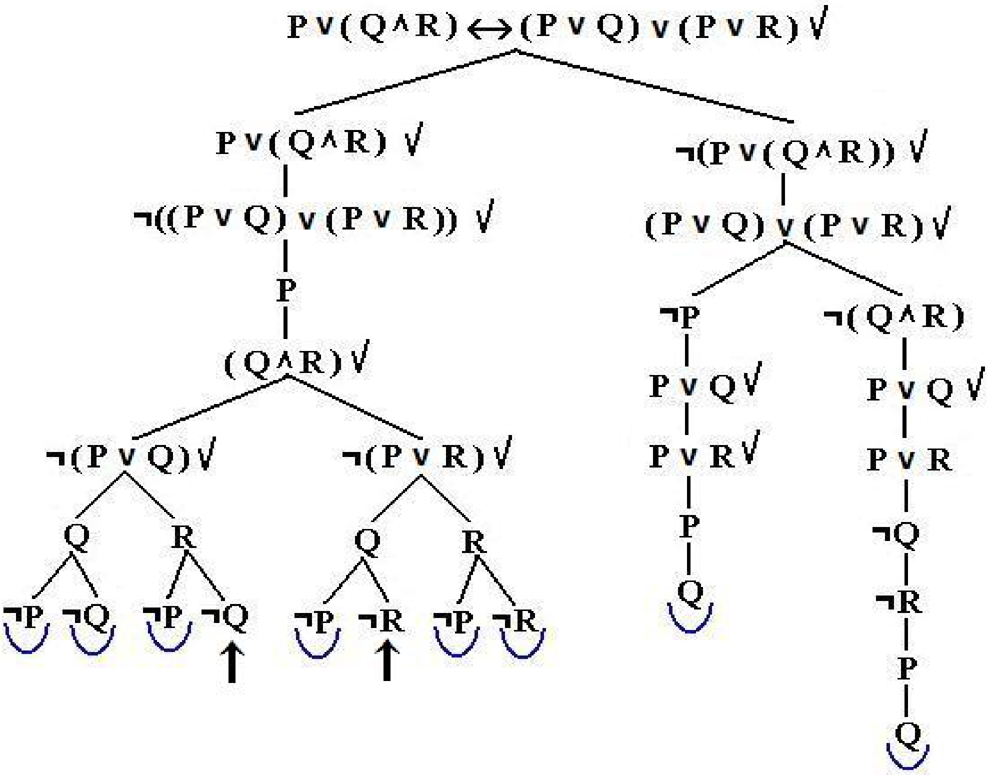


Figura 5.11: Exemplo com uma fórmula inválida em **LPC**

Capítulo 6

Um Sistema de Tablôs por Prova Direta para a Lógica Quantificacional Clássica

6.1 Definição do Sistema

Definiremos neste capítulo um *sistema de tablôs por prova direta* para a lógica quantificacional clássica, o qual será notado aqui por **STD***.

6.1.1 Notação: A partir de agora, no resto desta seção, **L** é uma linguagem para **LQC**, e **L**' é uma linguagem para **LQC**, obtida de **L** acrescentando uma infinidade de novas constantes.

6.1.2 Definição: **L** é *a linguagem inicial* de **STD***, e **L**' é *a linguagem de trabalho* de **STD***.

6.1.3 Definição: (**Eliminação de Quantificadores Vácuos**)
Dizemos que **P**' é obtido de **P** pela *eliminação de seus quantificadores vácuos* se cada subfórmula de **P** de uma das formas $\forall \boldsymbol{x}\, \mathbf{Q}$ ou $\exists \boldsymbol{x}\, \mathbf{Q}$, tal que $\boldsymbol{x}$ não é livre em **Q**, for substituída em **P** por **Q**.

6.1.4 Definição: *A função de inicialização de* **STD*** associa uma fórmula **P** de **L** a um tablô com um único nó não marcado cuja fórmula é obtida pela instanciação de todas as variáveis livres em **P** por novas constantes, e pela eliminação dos quantificadores vácuos de **P**.

6.1.5 Definição: Um *ramo é fechado em* **STD*** se ele possuir duas fórmulas contraditórias.

A coleção de regras de **STD*** possui, além das nove regras pertencentes ao sistema definido no capítulo anterior, mais quatro. Todas as nove regras de desenvolvimento de tablôs definidas no capítulo anterior continuam valendo aqui, com a diferença de que a linguagem de trabalho considerada aqui é uma linguagem para **LQC**. Por isso, iremos definir apenas as regras adicionais, que dizem respeito ao gerenciamento dos quantificadores "$\forall$" e "$\exists$".

Fórmula Universal ($\forall \boldsymbol{x}\,\mathbf{P}$): Esta regra associa a cada ramo contendo $\forall \boldsymbol{x}\,\mathbf{P}$ e à fórmula $\forall \boldsymbol{x}\,\mathbf{P}$ um único tablô com um único nó não marcado cuja fórmula é $\mathbf{P}(\boldsymbol{x}|\mathbf{c})$, obtida da instanciação da variável $\boldsymbol{x}$ por alguma constante $\mathbf{c}$ que ainda não figura no ramo considerado.

$$\begin{array}{c} | \\ \vdots \\ \forall \boldsymbol{x}\,\mathbf{P}\ \surd \\ \vdots \\ \mathbf{P}(\boldsymbol{x}|\mathbf{c}) \\ \text{com restrição} \end{array}$$

Figura 6.1: Fórmula Universal

Fórmula Existencial ($\exists \boldsymbol{x}\,\mathbf{P}$): Esta regra associa a cada ramo contendo $\exists \boldsymbol{x}\,\mathbf{P}$ e à fórmula $\exists \boldsymbol{x}\,\mathbf{P}$ um tablô com um único ramo com nós não marcados cujas fórmulas são $\mathbf{P}(\boldsymbol{x}|\mathbf{t}_1),...,\mathbf{P}(\boldsymbol{x}|\mathbf{t}_n)$, $\exists \boldsymbol{x}\,\mathbf{P}$, onde $\mathbf{P}(\boldsymbol{x}|\mathbf{t}_1),...,\mathbf{P}(\boldsymbol{x}|\mathbf{t}_n)$ são as instanciações de $\mathbf{P}$ por todos os termos fechados que figuram no ramo para os quais $\mathbf{P}$ ainda não foi instanciado.

$$\begin{array}{c} | \\ \vdots \\ \exists \boldsymbol{x}\,\mathbf{P}\ \surd \\ \vdots \\ \mathbf{P}(\boldsymbol{x}|\mathbf{t}_1) \\ \vdots \\ \mathbf{P}(\boldsymbol{x}|\mathbf{t}_n) \\ \vdots \\ \exists \boldsymbol{x}\,\mathbf{P} \end{array}$$

Figura 6.2: Fórmula Existencial

Negação de Fórmula Universal ($\neg\forall x\, \mathbf{P}$): Esta regra associa a cada ramo contendo $\neg\forall x\, \mathbf{P}$ e à fórmula $\neg\forall x\, \mathbf{P}$ um único tablô com um nó não marcado cuja fórmula é $\exists x \neg \mathbf{P}$.

$$\begin{array}{c} | \\ \vdots \\ \neg\forall x\, \mathbf{P}\ \surd \\ \vdots \\ \exists x \neg \mathbf{P} \end{array}$$

Figura 6.3: Negação de Fórmula Universal

Negação de Fórmula Existencial ($\neg\exists x\, \mathbf{P}$): Esta regra associa a cada ramo contendo $\neg\exists x\, \mathbf{P}$ e à fórmula $\neg\exists x\, \mathbf{P}$ um único tablô com um nó não marcado cuja fórmula é $\forall x \neg \mathbf{P}$.

$$\begin{array}{c} | \\ \vdots \\ \neg\exists x\, \mathbf{P}\ \surd \\ \vdots \\ \forall x \neg \mathbf{P} \end{array}$$

Figura 6.4: Negação de Fórmula Existencial

6.2 Prova da Correção de STD*

Mostraremos a seguir a correção de **STD*** com respeito a **LQC**, com base nas condições gerais de correção vistas no capítulo 4.

6.2.1 Definição: Seja σ um mundo tal que $\mathbf{c} \in D(\sigma)$. $\sigma(\mathbf{c}|\mathbf{d})$ é um mundo que só difere de σ no máximo quanto à atribuição da constante **c**, conforme as seguintes condições:

- $D(\sigma(\mathbf{c}|\mathbf{d})) = D(\sigma)$;
- $\sigma(\mathbf{c}|\mathbf{d})(\mathsf{s}) = \begin{cases} \sigma(\mathsf{s}), \text{ se } \mathsf{s} \neq \mathbf{c}; \\ \mathbf{d}, \text{ se } \mathsf{s} = \mathbf{c}. \end{cases}$

6.2.2 Definição: Se $\mathbf{I} = \langle \Delta, \sigma, \mathbf{s} \rangle$ é uma **LQC**-interpretação, então $\mathbf{I}(\mathbf{c}|\mathbf{d})$ é uma **LQC**-interpretação definida por $\mathbf{I}(\mathbf{c}|\mathbf{d}) = \langle \Delta, \sigma(\mathbf{c}|\mathbf{d}), \mathbf{s} \rangle$.

6.2.3 Lema: (Instanciação de variável por uma nova constante)
Se **c** não ocorre em **P**, então **P** é **LQC**-válido se, e somente se, $\mathbf{P}(\boldsymbol{x}|\mathbf{c})$ é **LQC**-válido.

Prova:

$(\Rightarrow)$

Suponha que **P** é **LQC**-válido, e seja **I** uma **LQC**-interpretação para $\mathbf{P}(\boldsymbol{x}|\mathbf{c})$.
Se $\boldsymbol{x}$ não é livre em **P**, então $\mathbf{P}(\boldsymbol{x}|\mathbf{c}) = \mathbf{P}$, e daí $\mathbf{I}_V(\mathbf{P}(\boldsymbol{x}|\mathbf{c})) = \mathbf{I}_V(\mathbf{P}) = \mathbf{v}$.
Se $\boldsymbol{x}$ é livre em **P**, temos, pelo lema 3.7.13, que $\mathbf{I}_V(\mathbf{P}(\boldsymbol{x}|\mathbf{c})) = \mathbf{I}(\boldsymbol{x}|\mathbf{I}_D(\mathbf{c}))_V(\mathbf{P})$.
Como **P** é **LQC**-válido, temos que $\mathbf{I}(\boldsymbol{x}|\mathbf{I}_D(\mathbf{c}))_V(\mathbf{P}) = \mathbf{v}$, donde $\mathbf{I}_V(\mathbf{P}(\boldsymbol{x}|\mathbf{c})) = \mathbf{v}$.
Portanto $\mathbf{P}(\boldsymbol{x}|\mathbf{c})$ é **LQC**-válido.

$(\Leftarrow)$

Suponha $\mathbf{P}(\boldsymbol{x}|\mathbf{c})$ é **LQC**-válido, e seja $\mathbf{I} = \langle \Delta, \sigma, \mathbf{s} \rangle$ uma **LQC**-interpretação para **P**.
Se $\boldsymbol{x}$ não é livre em **P**, então $\mathbf{P}(\boldsymbol{x}|\mathbf{c}) = \mathbf{P}$, e daí $\mathbf{I}_V(\mathbf{P}) = \mathbf{I}_V(\mathbf{P}(\boldsymbol{x}|\mathbf{c})) = \mathbf{v}$.
Caso $\boldsymbol{x}$ seja livre em **P**, considere $\mathbf{I'} = \mathbf{I}(\mathbf{c}|\mathbf{s}(\boldsymbol{x}))$.
Temos, conforme o lema 3.7.13, que $\mathbf{I'}_V(\mathbf{P}(\boldsymbol{x}|\mathbf{c})) = \mathbf{I'}(\boldsymbol{x}|\mathbf{I'}_D(\mathbf{c}))_V(\mathbf{P}) =$
$= \mathbf{I'}(\boldsymbol{x}|\mathbf{s}(\boldsymbol{x}))_V(\mathbf{P}) = \mathbf{I'}_V(\mathbf{P})$ (1).
Pelo lema 3.7.16, $\mathbf{I'}_V(\mathbf{P}) = \mathbf{I}_V(\mathbf{P})$ (2).
De (1), (2), e como $\mathbf{P}(\boldsymbol{x}|\mathbf{c})$ é **LQC**-válido, temos que $\mathbf{I}_V(\mathbf{P}) = \mathbf{v}$.
Portanto **P** é **LQC**-válido. **c.q.d.**

6.2.4 Corolário: (**Instanciação de variáveis por novas constantes**)
Se $\mathbf{c}_1,...,\mathbf{c}_n$ não ocorrem em **P**, então $\mathbf{P}(\boldsymbol{x}_1,...,\boldsymbol{x}_n|\mathbf{c}_1,...,\mathbf{c}_n)$ é **LQC**-válido se, e somente se, **P** é **LQC**-válido.

6.2.5 Lema: (**Eliminação dos quantificadores vácuos**)
Se **P**' é obtido de **P** pela eliminação de seus quantificadores vácuos, então $\models_{\mathbf{LQC}} \mathbf{P} \leftrightarrow \mathbf{P'}$.

O corolário da instanciação de variáveis por novas constantes tem um papel muito importante quando se trata do desenvolvimento de uma prova pelo método dos tablôs direto. Este corolário justifica a transformação de *fórmulas abertas*[1] em *fechadas*, evitando, assim, uma constante renomeação de variáveis no desenvolvimento de um tablô em **STD***. Este processo serve principalmente para evitar o choque de variáveis. Damos

[1]Fórmulas que possuem variáveis livres.

abaixo um exemplo da instanciação de variáveis livres por novas constantes:

$$\exists x \forall y\ \mathbf{p}(x,y,\ \mathbf{f}(z),\ \mathbf{g}(w,z)) \rightarrow \forall y \exists x\ \mathbf{p}(x,y,\ \mathbf{f}(z),\ \mathbf{g}(w,z))$$
$$\Downarrow$$
$$\exists x \forall y\ \mathbf{p}(x,y,\ \mathbf{f}(\mathbf{c}_1),\ \mathbf{g}(\mathbf{c}_2,\ \mathbf{c}_1)) \rightarrow \forall y\ \exists x\ \mathbf{p}(x,y,\ \mathbf{f}(\mathbf{c}_1),\ \mathbf{g}(\mathbf{c}_2,\ \mathbf{c}_1))$$

6.2.6 Lema: Se o tablô inicial para **P** em **STD*** é **LQC**-válido, então **P** é **LQC**-válido.

Prova:

Seja $\mathbf{T}_0$ o tablô inicial para **P** em **STD***.

Por definição, $\mathbf{T}_0$ possui um único nó cujo conteúdo é **P'**, onde **P'** é obtido instanciando todas as suas variáveis livres por novas constantes, e eliminando todos os seus quantificadores vácuos.

Se $\mathbf{T}_0$ é **LQC**-válido em **STD***, então todos os ramos de $\mathbf{T}_0$ são **LQC**-válidos, e daí o seu único ramo cuja coleção de fórmulas é $\{$**P'**$\}$ é **LQC**-válido, donde **P'** é **LQC**-válido, e daí, pelos lemas 6.2.3 e 6.2.5, temos que **P** é **LQC**-válido. **c.q.d.**

O lema seguinte é análogo ao lema 5.2.3, porém é concernente à semântica de **LQC**.

6.2.7 Lema: As seguintes proposições são verdadeiras:

(i) Se $\mathbf{P} \rightarrow \mathbf{Q} \in \Gamma$ e $\Gamma \cup \{\neg\mathbf{P}, \mathbf{Q}\}$ é **LQC**-válido, então Γ é **LQC**-válido.

(ii) Se $\mathbf{P} \wedge \mathbf{Q} \in \Gamma$, $\Gamma \cup \{\mathbf{P}\}$ e $\Gamma \cup \{\mathbf{Q}\}$ são **LQC**-válidos, então Γ é **LQC**-válido.

(iii) Se $\mathbf{P} \vee \mathbf{Q} \in \Gamma$ e $\Gamma \cup \{\mathbf{P},\mathbf{Q}\}$ é **LQC**-válido, então Γ é **LQC**-válido.

(iv) Se $\neg(\mathbf{P} \rightarrow \mathbf{Q}) \in \Gamma$ e $\Gamma \cup \{\mathbf{P}\}$ e $\Gamma \cup \{\neg\mathbf{Q}\}$ são **LQC**-válidos, então Γ é **LQC**-válido.

(v) Se $\neg(\mathbf{P} \wedge \mathbf{Q}) \in \Gamma$ e $\Gamma \cup \{\neg\mathbf{P}, \neg\mathbf{Q}\}$ é **LQC**-válido, então Γ é **LQC**-válido.

(vi) Se $\neg(\mathbf{P} \vee \mathbf{Q}) \in \Gamma$, $\Gamma \cup \{\neg\mathbf{P}\}$ e $\Gamma \cup \{\neg\mathbf{Q}\}$ são **LQC**-válidos, então Γ é **LQC**-válido.]

(vii) Se $\neg\neg\mathbf{P} \in \Gamma$ e $\Gamma \cup \{\mathbf{P}\}$ é **LQC**-válido, então Γ é **LQC**-válido.

(viii) Se $\mathbf{P} \leftrightarrow \mathbf{Q} \in \Gamma$, $\Gamma \cup \{\mathbf{P},\neg\mathbf{Q}\}$ e $\Gamma \cup \{\neg\mathbf{P},\mathbf{Q}\}$ são **LQC**-válidos, então Γ é **LQC**-válido.

(ix) Se $\neg(\mathbf{P} \leftrightarrow \mathbf{Q}) \in \Gamma$, $\Gamma \cup \{\mathbf{P}, \mathbf{Q}\}$ e $\Gamma \cup \{\neg\mathbf{P}, \neg\mathbf{Q}\}$ são **LQC**-válidos, então Γ é **LQC**-válido.

Prova: É análoga à prova do lema 5.2.3, porém é concernente à semântica de **LQC**. **c.q.d.**

6.2.8 Lema: Se **c** é uma constante não ocorrendo em **P**, então as seguintes proposições são equivalentes:

- $\mathbf{I}_V(\forall \boldsymbol{x}\,\mathbf{P}) = \mathbf{v}$;
- $\mathbf{I}(\mathbf{c}|\mathbf{d})_V(\mathbf{P}(\boldsymbol{x}|\mathbf{c})) = \mathbf{v}$, para todo $\mathbf{d} \in \Delta$.

6.2.9 Lema: As seguintes proposições são válidas:

(i) Se $\forall \boldsymbol{x}\,\mathbf{P} \in \Gamma$, **c** não figura em Γ e $\Gamma \cup \{\mathbf{P}(\boldsymbol{x}|\mathbf{c})\}$ é **LQC**-válido, então Γ é **LQC**-válido.

(ii) Se $\exists \boldsymbol{x}\,\mathbf{P} \in \Gamma$, $\mathbf{t}_1,\ldots,\mathbf{t}_n$ são termos fechados e $\Gamma \cup \{\mathbf{P}(\boldsymbol{x}|\mathbf{t}_1),\ldots,\mathbf{P}(\boldsymbol{x}|\mathbf{t}_n)\}$ é **LQC**-válido, então Γ é **LQC**-válido.

(iii) $\mathbf{I}_V(\neg\forall \boldsymbol{x}\,\mathbf{P}) = \mathbf{I}_V(\exists \boldsymbol{x}\neg\mathbf{P})$.

(iv) $\mathbf{I}_V(\neg\exists \boldsymbol{x}\,\mathbf{P}) = \mathbf{I}_V(\forall \boldsymbol{x}\neg\mathbf{P})$.

Prova de (i):

Suponha que "$\forall \boldsymbol{x}\,\mathbf{P}$" $\in \Gamma$, $\Gamma \cup \{\mathbf{P}(\boldsymbol{x}|\mathbf{c})\}$ é **LQC**-válido, onde **c** não figura em Γ.

Seja $\mathbf{I} = \langle \Delta, \sigma, \mathbf{s}\rangle$ uma **LQC**-interpretação.

Para cada $\mathbf{d} \in \Delta$, considere $\begin{cases} \sigma_d = \sigma(\mathbf{c}|\mathbf{d}) \\ \mathbf{I}_d = \mathbf{I}(\mathbf{c}|\mathbf{d}) \end{cases}$

$\mathbf{I}_V(\forall \boldsymbol{x}\,\mathbf{P}) = \min\{\mathbf{I}(\boldsymbol{x}|\mathbf{d})_V(\mathbf{P}) \mid \mathbf{d} \in \Delta\} = \min\{\mathbf{I}(\boldsymbol{x}|\sigma_d(\mathbf{c}))_V(\mathbf{P}) \mid \mathbf{d} \in \Delta\}$, daí pelo lema 3.7.16, $\mathbf{I}_V(\forall \boldsymbol{x}\,\mathbf{P}) = \min\{\mathbf{I}_d(\boldsymbol{x}|\sigma_d(\mathbf{c}))_V(\mathbf{P}) \mid \mathbf{d} \in \Delta\}$.

Como $\sigma(\mathbf{c}) = (\mathbf{I}_d)_D(\mathbf{c})$, e pelo lema 3.7.13, segue-se que $\mathbf{I}_V(\forall \boldsymbol{x}\,\mathbf{P}) = \min\{(\mathbf{I}_d)_V(\mathbf{P}(\boldsymbol{x}|\sigma_d(\mathbf{c})) \mid \mathbf{d} \in \Delta\}$.

Como $\Gamma \cup \{\mathbf{P}(\boldsymbol{x}|\mathbf{c})\}$ é **LQC**-válido, temos que $(\mathbf{I}_d)_V(\mathbf{P}(\boldsymbol{x}|\mathbf{c})) = \mathbf{v}$, para todo $\mathbf{d} \in \Delta$, donde $\mathbf{I}_V(\forall \boldsymbol{x}\,\mathbf{P}) = \mathbf{v}$, e portanto Γ é **LQC**-válido.

Prova de (ii):

Suponha que "$\exists \boldsymbol{x}\,\mathbf{P}$" $\in \Gamma$ e $\Gamma \cup \{\mathbf{P}(\boldsymbol{x}|\mathbf{t}_1),\ldots,\mathbf{P}(\boldsymbol{x}|\mathbf{t}_n)\}$ é **LQC**-válido, onde $\mathbf{t}_1,\ldots,\mathbf{t}_n$ são termos fechados.

Se $\mathbf{n} = 0$, então $\Gamma \cup \{\mathbf{P}(\boldsymbol{x}|\mathbf{t}_1),\ldots,\mathbf{P}(\boldsymbol{x}|\mathbf{t}_n)\} = \Gamma$, e daí Γ é **LQC**-válido, e daí iremos supor, no resto desta prova, que $\mathbf{n} > 0$.

Dada uma **LQC**-interpretação $\mathbf{I} = \langle \Delta,\sigma,\mathbf{s}\rangle$, temos que **I** satisfaz pelo menos uma fórmula **R** de $\Gamma \cup \{\mathbf{P}(\boldsymbol{x}|\mathbf{t}_1),\ldots,\mathbf{P}(\boldsymbol{x}|\mathbf{t}_n)\}$.

Se $\mathbf{R} \in \Gamma$, então **I** obviamente satisfaz pelo menos uma fórmula de Γ.

Se $\mathbf{R} \notin \Gamma$, então $\mathbf{R}$ é $\mathbf{P}(\boldsymbol{x}|\mathbf{t}_i)$, para algum i $\in \{1,...,n\}$, donde $\mathbf{I}_V(\mathbf{P}(\boldsymbol{x}|\mathbf{t}_i))$ = **v**, e daí pelo lema 3.7.13, temos que $\mathbf{I}(\boldsymbol{x}|\mathbf{I}_D(\mathbf{t}_i))_V(\mathbf{P})$, e daí $\mathbf{I}_V(\exists \boldsymbol{x}\, \mathbf{P})$ = **v**, e portanto Γ é **LQC**-válido.

Prova de (iii):
$\mathbf{I}_V(\neg\forall \boldsymbol{x}\, \mathbf{P}) \neq \mathbf{I}_V(\forall \boldsymbol{x}\, \mathbf{P})$. (1)
$\mathbf{I}_V(\forall \boldsymbol{x}\, \mathbf{P}) = \min\{\mathbf{I}(\boldsymbol{x}|\mathbf{d})_V(\mathbf{P}) \mid \mathbf{d} \in \Delta\}$. (2)
$\mathbf{I}_V(\exists \boldsymbol{x} \neg \mathbf{P}) = \max\{\mathbf{I}(\boldsymbol{x}|\mathbf{d})_V(\neg\mathbf{P}) \mid \mathbf{d} \in \Delta\}$. (3)
$\min\{\mathbf{I}(\boldsymbol{x}|\mathbf{d})_V(\mathbf{P}) \mid \mathbf{d} \in \Delta\} \neq \max\{\mathbf{I}(\boldsymbol{x}|\mathbf{d})_V(\neg\mathbf{P}) \mid \mathbf{d} \in \Delta\}$. (4)
De (2), (3) e (4), temos que $\mathbf{I}_V(\exists \boldsymbol{x} \neg \mathbf{P}) \neq \mathbf{I}_V(\forall \boldsymbol{x}\, \mathbf{P})$. (5)
De (1) e (5), temos que $\mathbf{I}_V(\neg\forall \boldsymbol{x}\, \mathbf{P}) = \mathbf{I}_V(\exists \boldsymbol{x} \neg \mathbf{P})$.

Prova de (iv):
$\mathbf{I}_V(\neg\exists \boldsymbol{x}\, \mathbf{P}) \neq \mathbf{I}_V(\exists \boldsymbol{x}\, \mathbf{P})$. (1)
$\mathbf{I}_V(\exists \boldsymbol{x}\, \mathbf{P}) = \max\{\mathbf{I}(\boldsymbol{x}|\mathbf{d})_V(\mathbf{P}) \mid \mathbf{d} \in \Delta\}$. (2)
$\mathbf{I}_V(\forall \boldsymbol{x} \neg \mathbf{P}) = \min\{\mathbf{I}(\boldsymbol{x}|\mathbf{d})_V(\neg\mathbf{P}) \mid \mathbf{d} \in \Delta\}$. (3)
$\max\{\mathbf{I}(\boldsymbol{x}|\mathbf{d})_V(\mathbf{P}) \mid \mathbf{d} \in \Delta\} \neq \min\{\mathbf{I}(\boldsymbol{x}|\mathbf{d})_V(\neg\mathbf{P}) \mid \mathbf{d} \in \Delta\}$. (4)
De (2), (3) e (4), temos que $\mathbf{I}_V(\exists \boldsymbol{x}\, \mathbf{P}) \neq \mathbf{I}_V(\forall \boldsymbol{x} \neg \mathbf{P})$. (5)
De (1) e (5), temos que $\mathbf{I}_V(\neg\exists \boldsymbol{x}\, \mathbf{P}) = \mathbf{I}_V(\forall \boldsymbol{x} \neg \mathbf{P})$. **c.q.d.**

6.2.10 Lema: Dada uma seqüência $(\mathbf{T}_n)_{n\in I}$ de desenvolvimento de tablôs em **STD***, para cada **n** $\in$ I, se **n** $>$ 0 e $\mathbf{T}_n$ é **LQC**-válido, então $\mathbf{T}_{n-1}$ é **LQC**-válido.
Prova:
É análoga à prova do lema 5.2.4, porém usando os lemas 6.2.7 e 6.2.9 no lugar do lema 5.2.4. **c.q.d.**

6.2.11 Lema: Todo tablô fechado em **STD*** é **LQC**-válido.
Prova:
É análoga à prova do lema 5.2.5, porém é concernente à semântica de **LQC**. **c.q.d.**

6.2.12 Teorema: STD* é correto com respeito a **LQC**.
Prova:
Basta aplicar os lemas 6.2.6, 6.2.10 e 6.2.11. **c.q.d.**

6.2.13 Corolário: Se existe um tablô direto fechado para **P** em **STD***, então **P** é **LQC**-válido.
Prova:
Basta aplicar o teorema 6.2.12 e a definição de correção de um sistema de

tablôs com respeito a uma dada lógica. **c.q.d.**

6.3 Prova da Completude de STD*

A prova da completude para **STD*** com respeito a **LQC** segue as condições gerais de completude estabelecidas no capítulo 4.

6.3.1 Lema: Todo ramo exaurido aberto em **STD*** é **LQC**-inválido.
Prova:
Para provar este lema, basta mostrar que existe uma interpretação cuja valoração atribui o valor veritativo **f** as fórmulas de um ramo exaurido aberto. Seja ρ um ramo exaurido aberto em **STD*** e Γ a sua coleção de fórmulas. Seja **L** a menor linguagem para **LQC** que contém todas as constantes, sinais funcionais e sinais predicativos em Γ.
Seja $\mathbf{I} = \langle \Delta, \sigma, \mathbf{s} \rangle$ uma **LQC**-interpretação para **L** definida por:
(i) Δ é a coleção de todos os termos fechados em **L**;
(ii) $\sigma(\mathbf{c})$ = "**c**";
(iii) $\sigma(\mathbf{f})(\mathbf{t}_1,...,\mathbf{t}_n)$ = "$\mathbf{f}(\mathbf{t}_1,...,\mathbf{t}_n)$";
(iv) $\langle \mathbf{t}_1,...,\mathbf{t}_n \rangle \in \sigma(\mathbf{p})$ se, somente se, "$\mathbf{p}(\mathbf{t}_1,...,\mathbf{t}_n)$" $\notin \Gamma$;
(v) $\mathbf{s}(\boldsymbol{x})$ = "$\boldsymbol{x}$", para qualquer variável $\boldsymbol{x}$.
Provaremos, por indução sobre o número de ocorrências de conectivos e quantificadores em **P**, as seguintes propriedades.
- Se $\mathbf{P} \in \Gamma$, então $\mathbf{I}_V(\mathbf{P}) = \mathbf{f}$.
- Se $\neg\mathbf{P} \in \Gamma$, então $\mathbf{I}_V(\mathbf{P}) = \mathbf{v}$.

Caso i) **P** é uma fórmula atômica da forma $\mathbf{p}(\mathbf{t}_1,...,\mathbf{t}_n)$:
- Se $\mathbf{P} \in \Gamma$, então $\mathbf{p}(\mathbf{t}_1,...,\mathbf{t}_n) \in \Gamma$.
  Daí, por (iv), $\langle \mathbf{t}_1,...,\mathbf{t}_n \rangle \notin \sigma(\mathbf{p})$.
  Logo $\mathbf{I}_V(\mathbf{p}(\mathbf{t}_1,...,\mathbf{t}_n)) = \mathbf{f}$, portanto $\mathbf{I}_V(\mathbf{P}) = \mathbf{f}$.
- Se $\neg\mathbf{P} \in \Gamma$, então $\neg\mathbf{p}(\mathbf{t}_1,...,\mathbf{t}_n) \in \Gamma$, e daí $\mathbf{p}(\mathbf{t}_1,...,\mathbf{t}_n) \notin \Gamma$.
  Daí, por (iv), $\langle \mathbf{t}_1,...,\mathbf{t}_n \rangle \in \sigma(\mathbf{p})$.
  Daí $\mathbf{I}_V(\mathbf{p}(\mathbf{t}_1,...,\mathbf{t}_n)) = \mathbf{v}$, e daí $\mathbf{I}_V(\mathbf{p}(\mathbf{t}_1,...,\mathbf{t}_n)) = \mathbf{v}$.
  Logo $\mathbf{I}_V(\neg\mathbf{p}(\mathbf{t}_1,...,\mathbf{t}_n)) = \mathbf{f}$, portanto $\mathbf{I}_V(\neg\mathbf{P}) = \mathbf{f}$.

Caso ii) **P** é de uma das formas $\mathbf{Q} \rightarrow \mathbf{R}$, ou $\mathbf{Q} \wedge \mathbf{R}$, ou $\mathbf{Q} \vee \mathbf{R}$, ou $\neg(\mathbf{Q} \rightarrow \mathbf{R})$, ou $\neg(\mathbf{Q} \wedge \mathbf{R})$, ou $\neg(\mathbf{Q} \vee \mathbf{R})$, $\neg\neg\mathbf{Q}$, ou $\mathbf{Q} \leftrightarrow \mathbf{R}$, $\neg(\mathbf{Q} \leftrightarrow \mathbf{R})$.
Para este caso, a prova é análoga ao trecho correspondente da prova do lema 5.3.1.

Caso iii) **P** é da forma $\forall \boldsymbol{x}\, \mathbf{Q}$:

- Se **P** $\in \Gamma$, então "$\forall \boldsymbol{x}\, \mathbf{Q}$" $\in \Gamma$, e daí existe uma constante **c** não figurando em **Q** tal que $\mathbf{Q}(\boldsymbol{x}|\mathbf{c}) \in \Gamma$.
  Por hipótese de indução e pelo lema 3.7.13 $\mathbf{I}_V(\mathbf{Q}(\boldsymbol{x}|\mathbf{c})) = \mathbf{I}(\boldsymbol{x}|\mathbf{c})_V(\mathbf{Q})$, donde $\mathbf{I}_V(\forall \boldsymbol{x}\, \mathbf{Q}) = \mathbf{f}$, portanto $\mathbf{I}_V(\mathbf{P}) = \mathbf{f}$.
- Se $\neg\mathbf{P} \in \Gamma$, então "$\neg\forall \boldsymbol{x}\, \mathbf{Q}$" $\in \Gamma$, daí $\exists \boldsymbol{x} \neg \mathbf{Q} \in \Gamma$, donde, para qualquer termo fechado **t** em Γ, $\mathbf{Q}(\boldsymbol{x}|\mathbf{t})$.
  Por hipótese de indução, temos que, para qualquer termos fechados **t** em Γ $\mathbf{I}_V(\mathbf{Q}(\boldsymbol{x}|\mathbf{t})) = \mathbf{v}$, e daí pelo lema 3.7.13, $\mathbf{I}(\boldsymbol{x}|\mathbf{t})_V(\mathbf{Q}) = \mathbf{v}$, para qualquer termo fechado **t** em Γ, e portanto $\mathbf{I}_V(\forall \boldsymbol{x}\, \mathbf{Q}) = \mathbf{v}$, ou seja, $\mathbf{I}_V(\mathbf{P}) = \mathbf{v}$.

Caso iv) **P** é da forma $\exists \boldsymbol{x}\, \mathbf{Q}$:

- Se **P** $\in \Gamma$, então "$\exists \boldsymbol{x}\, \mathbf{Q}$" $\in \Gamma$, e daí $\mathbf{Q}(\boldsymbol{x}|\mathbf{t}) \in \Gamma$, para todo termo fechado **t** em Γ.
  Por hipótese de indução, $\mathbf{I}_V(\mathbf{Q}(\boldsymbol{x}|\mathbf{t})) = \mathbf{f}$, para todo termo fechado **t** em Γ. Logo, $\mathbf{I}_V(\exists \boldsymbol{x}\, \mathbf{Q}) = \mathbf{f}$, ou seja, $\mathbf{I}_V(\mathbf{P}) = \mathbf{f}$.
- Se $\neg\mathbf{P} \in \Gamma$, então "$\neg\exists \boldsymbol{x}\, \mathbf{Q}$" $\in \Gamma$, daí $\forall \boldsymbol{x} \neg \mathbf{Q} \in \Gamma$, donde para alguma constante **c** não figurando em **Q**, $\neg\mathbf{Q}(\boldsymbol{x}|\mathbf{c}) \in \Gamma$.
  Por hipótese de indução, temos que $\mathbf{I}_V(\mathbf{Q}(\boldsymbol{x}|\mathbf{t})) = \mathbf{v}$, donde, pelo lema 3.7.13, temos que $\mathbf{I}(\boldsymbol{x}|\mathbf{c})_V(\mathbf{Q}) = \mathbf{v}$, e portanto $\mathbf{I}_V(\exists \boldsymbol{x}\, \mathbf{Q}) = \mathbf{v}$, ou seja, $\mathbf{I}_V(\mathbf{P}) = \mathbf{v}$.

Caso v) **P** é da forma $\neg\forall \boldsymbol{x}\, \mathbf{Q}$:

- Se **P** $\in \Gamma$, então "$\neg\forall \boldsymbol{x}\, \mathbf{Q}$" $\in \Gamma$, daí segundo o raciocínio feito na prova da segunda proposição para o caso (**iii**), temos que $\mathbf{I}_V(\forall \boldsymbol{x}\, \mathbf{Q}) = \mathbf{v}$, daí $\mathbf{I}_V(\neg\forall \boldsymbol{x}\, \mathbf{Q}) = \mathbf{f}$, e portanto $\mathbf{I}_V(\mathbf{P}) = \mathbf{f}$.
- Se $\neg\mathbf{P} \in \Gamma$, "$\neg\neg\forall \boldsymbol{x}\, \mathbf{Q}$" $\in \Gamma$, daí $\forall \boldsymbol{x}\, \mathbf{Q} \in \Gamma$, donde, segundo o raciocínio feito na prova da primeira proposição para o caso (**iii**), temos que $\mathbf{I}_V(\forall \boldsymbol{x}\, \mathbf{Q}) = \mathbf{f}$, daí $\mathbf{I}_V(\neg\forall \boldsymbol{x}\, \mathbf{Q}) = \mathbf{v}$, e portanto $\mathbf{I}_V(\mathbf{P}) = \mathbf{v}$.

Caso vi) **P** é da forma $\neg\exists \boldsymbol{x}\, \mathbf{Q}$:

- Se **P** $\in \Gamma$, então "$\neg\exists \boldsymbol{x}\, \mathbf{Q}$" $\in \Gamma$, daí segundo o raciocínio feito na prova da segunda proposição para o caso (**iv**), temos que $\mathrm{I}_V(\exists \boldsymbol{x}\, \mathbf{Q}) = \mathbf{v}$, daí $\mathbf{I}_V(\neg\exists \boldsymbol{x}\, \mathbf{Q}) = \mathbf{f}$, e portanto $\mathbf{I}_V(\mathbf{P}) = \mathbf{f}$.
- Se $\neg\mathbf{P} \in \Gamma$, então "$\neg\neg\exists \boldsymbol{x}\, \mathbf{Q}$" $\in \Gamma$, daí $\exists \boldsymbol{x}\, \mathbf{Q} \in \Gamma$, donde, segundo o raciocínio feito na prova da primeira proposição para o caso (**iv**), temos que $\mathbf{I}_V(\exists \boldsymbol{x}\, \mathbf{Q}) = \mathbf{f}$, daí $\mathbf{I}_V(\neg\exists \boldsymbol{x}\, \mathbf{Q}) = \mathbf{v}$, e portanto $\mathbf{I}_V(\mathbf{P}) = \mathbf{v}$. **c.q.d.**

6.3.2 Lema: Se **P** é **LQC**-válido, então o tablô inicial para **P** é **LQC**-válido.

Prova:

Suponha que **P** é **LQC**-válido, e seja $\mathbf{T}_0$ o tablô inicial para **P**.

Por definição, $\mathbf{T}_0$ possui um único nó cuja coleção de formulas é $\{\mathbf{P'}\}$, onde **P'** é obtido de **P** instanciando todas as suas variáveis livres por constantes e eliminando todos os seus quantificadores vácuos.

Como **P** é **LQC**-válido, temos que, pelo corolário 6.2.4 e pelo lema 6.2.5, **P'** é **LQC**-válido, donde $\mathbf{T}_0$ é **LQC**-válido. **c.q.d.**

6.3.3 Teorema: **STD*** é completo com respeito a **LQC**.
Prova:
Basta aplicar os lemas 6.3.1 e 6.3.2. **c.q.d.**

6.3.4 Corolário: Se **P** é **LQC**-válido, então toda seqüência de desenvolvimento completa de tablôs para **P** em **STD*** termina com um tablô fechado para **P** em **STD***.
Prova:
Basta aplicar o teorema 6.3.3 e a definição de completude de um sistema de tablôs com respeito a uma dada lógica. **c.q.d.**

6.3.5 Corolário: As seguintes proposições são equivalentes:
- **P** é **LQC**-válido;
- toda seqüência de desenvolvimento de tablôs para **P** em **STD*** termina com um tablô fechado para **P** em **STD***;
- existe um tablô fechado para **P** em **STD***.

6.4 Exemplos

6.4.1 Notação: Usaremos também aqui, da mesma forma que foi especificada na página 55, o sinal "$\smile$" para indicar quando um dado ramo é fechado em **STD***, e o sinal "$\uparrow$" para dizer que um ramo é aberto em **STD***.

6.4.2 Exemplo com uma fórmula válida em LQC
O desenvolvimento de um tablô por prova direta, exposto na figura 6.5, para a fórmula "$\forall \boldsymbol{x}(\mathbf{p}\boldsymbol{x} \to \mathbf{q}\boldsymbol{x}) \to (\forall \boldsymbol{x}\, \mathbf{p}\boldsymbol{x} \to \forall \boldsymbol{x}\, \mathbf{q}\boldsymbol{x})$" não obedeceu nenhuma ordem pré-definida, porém fez-se uso de uma ordem mais adequada para obter o fechamento dos ramos mais rapidamente. Assim, obtivemos um tablô fechado para a referida fórmula, portanto temos que "$\forall \boldsymbol{x}(\mathbf{p}\boldsymbol{x} \to \mathbf{q}\boldsymbol{x}) \to (\forall \boldsymbol{x}\, \mathbf{p}\boldsymbol{x} \to \forall \boldsymbol{x}\, \mathbf{q}\boldsymbol{x})$" é uma tese em **LQC**, ou seja, $\models_{\mathbf{LQC}} \forall \boldsymbol{x}(\mathbf{p}\boldsymbol{x} \to \mathbf{q}\boldsymbol{x}) \to (\forall \boldsymbol{x}\, \mathbf{p}\boldsymbol{x} \to \forall \boldsymbol{x}\, \mathbf{q}\boldsymbol{x})$.

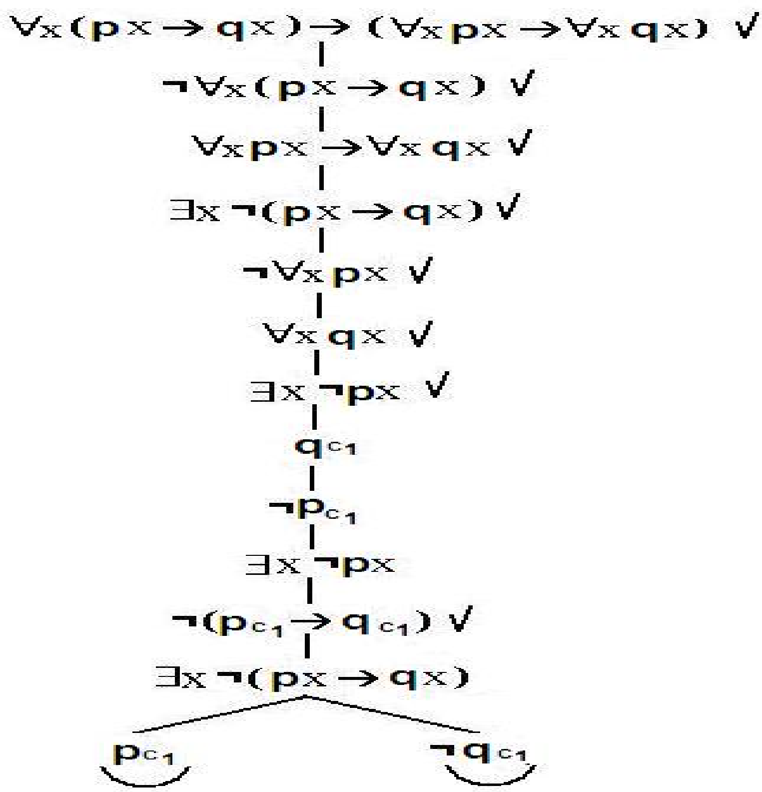


Figura 6.5: Exemplo com uma fórmula válida em **LQC**

6.4.3 Exemplo com uma fórmula inválida em LQC

Cada ramo exaurido aberto de um tablô para uma fórmula em **STD*** dá uma interpretação para a qual todas as fórmulas do ramo são falsas, e portanto a fórmula inicial não é **LQC**-válida. A figura 6.6 exibe o tablô final de uma seqüência de desenvolvimento completa em **STD***, o qual possui dois ramos exauridos abertos. Daí podemos concluir, com base no teorema 6.3.3, que a fórmula "$\exists \boldsymbol{x}(\mathbf{p}\boldsymbol{x} \rightarrow \mathbf{q}\boldsymbol{x}) \rightarrow (\exists \boldsymbol{x}\, \mathbf{p}\boldsymbol{x} \rightarrow \forall \boldsymbol{x}\, \mathbf{q}\boldsymbol{x})$" não é válida em **LQC**, isto é, $\not\models_{\mathbf{LQC}} \exists \boldsymbol{x}(\mathbf{p}\boldsymbol{x} \rightarrow \mathbf{q}\boldsymbol{x}) \rightarrow (\exists \boldsymbol{x}\, \mathbf{p}\boldsymbol{x} \rightarrow \forall \boldsymbol{x}\, \mathbf{q}\boldsymbol{x})$.

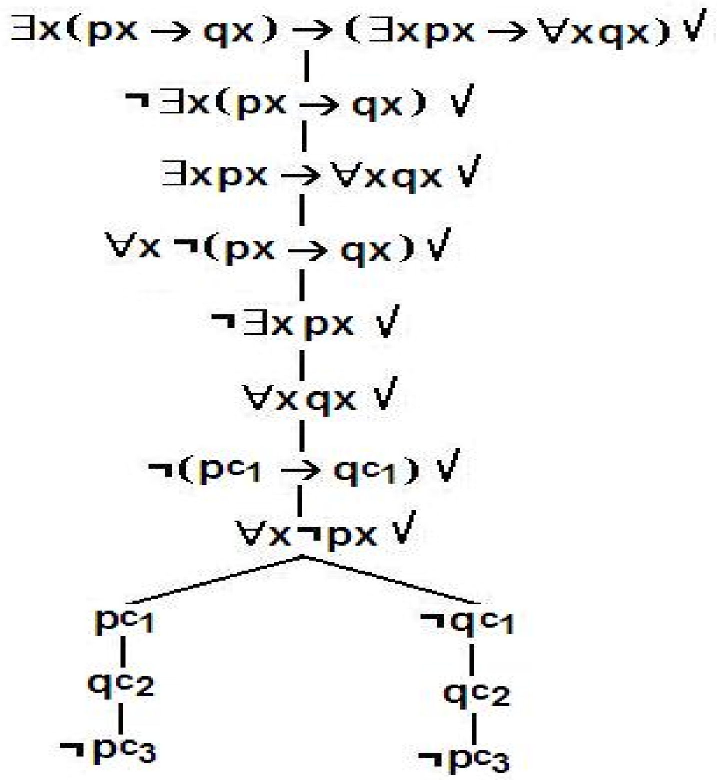


Figura 6.6: Exemplo com uma fórmula inválida em **LQC**

Capítulo 7

Considerações Finais

No presente trabalho foram dadas certas condições gerais que um sistema de tablôs, por prova direta, deve satisfazer para que o mesmo seja correto e completo com respeito a uma dada lógica munida de uma semântica de valorações.

Construímos também sistemas de tablôs para a lógica clássica nos níveis proposicional e quantificacional, segundo o método de prova direta, satisfazendo as condições gerais de correção e completude.

A nossa principal motivação reside no fato de que todas as referências por nós consultadas classificam o método dos tablôs (e o método da resolução) como um procedimento de prova por refutação, o que descobrimos não ser verdade.

Em "**Noções Básicas de Teoria dos Conjuntos**" foram especificados alguns conceitos que possibilitaram uma definição formal de árvore e seus elementos. Através dessa abordagem foi possível fixar um tablô como uma árvore de fórmulas.

Em "**A Lógica Clássica**" vimos conceitos relacionados a linguagens e a semânticas de valorações, direcionados posteriormente para os níveis proposicional e quantificacional da lógica clássica.

Em "**O Método dos Tablôs**" foram definidos sistemas de tablôs em um ambiente abstrato, ou seja, de forma independente do domínio de aplicação. Posteriormente mostramos como tal idéia pode ser utilizada para a concepção de sistemas de tablôs condizentes com a prova direta. Nos dois capítulos seguintes, aplicamos este método para a definição de sistemas de tablôs diretos para as lógicas proposicional e quantificacional clássica, e mostramos que ambos satisfazem as condições gerais de correção e completude formuladas anteriormente.

Esperamos que este trabalho tenha rompido com o dogma implícito na crença da comunidade científica e acadêmica de que o método dos tablôs é vinculado à refutação.

Entre as linhas futuras de pesquisa que podem provir deste trabalho, destacamos as seguintes:

- As condições gerais de correção e completude aqui apresentadas são suficientes. Uma questão imediata que surge é se as mesmas são também necessárias.
- Divisamos a possibilidade de construção de sistemas de tablôs por prova direta para diversas lógicas não clássicas, entre elas certas lógicas paraconsistentes e/ou paracompletas.
- Condições gerais de correção e completude também podem ser formuladas e demonstradas para sistemas de tablôs baseados em refutação.
- É possível definir, de forma abstrata, sistemas de resolução, bem como formular e demonstrar condições gerais de correção e completude dos mesmos, tanto para prova direta como para refutação.
- É possível construir sistemas de resolução por prova direta ou por refutação, tanto para a lógica clássica, como para diversas lógicas não clássicas.
- Diversas estratégias que incrementam a eficiência de **STD** e de **STD*** podem ser adotadas.

Referências Bibliográficas

[1] ABE, J. M.; SCALZITTI, A. & INÁCIO DA SILVA FILHO, J. *Introdução à Lógica para a Ciência da Computação*. Editora Arte e Ciência, São Paulo, 2002.

[2] BARRETO, J. M. *Inteligência Artificial no Limiar do Século XXI*. Editora da UFSC, Florianópolis - SC, 1999.

[3] BELL, J. & MACHOVER, M. *A Course in Mathematical Logic*. North-Holland, 1977.

[4] BESSIE, J. & GLENNAN, S. *Elements of deductive inference, an introduction to symbolic logic*. Wadsworth Publishing Company, Davis Drive Belmont, 1999.

[5] BETH, E. W. *Semantic Entailment and Formal Derivability*. Mededelingen van de Koninklijke Nederlandse Akademie van Wetenschappen, Amsterdam, 1955.

[6] BUCHSBAUM, A. & ARRUDA, K. R. Lógica Aplicada. Universidade Federal do Ceará. Apostila, 1995.

[7] BUCHSBAUM, A. & PEQUENO, T. Um provador paraconsistente. *Anais do V Simpósio Brasileiro de Inteligência Artificial,* (1988).

[8] BUCHSBAUM, A. & PEQUENO, T. O método dos tableaux generalizado e sua aplicação ao raciocínio automático em lógicas não clássi-

cas. *Departamento de Informática da PUC-Rio,* (Setembro de 1990). Revista “O que nos faz pensar”.

[9] BUCHSBAUM, A. & PEQUENO, T. A reasoning method for a Paraconsistent logic. *Studia Logic nº 52/2,* (maio de 1993).

[10] BUCHSBAUM, A. & PEQUENO, T. Uma família de lógicas paraconsistentes e/ou paracompletas com semânticas recursivas. *Coleção Documentos, Série Lógica e Teoria da Ciência, nº 14,* (Setembro de 1993).

[11] BUCHSBAUM, A. & PEQUENO, T. Automated deduction with non classical negations. *Proceedings of 3rd Workshop on Theorem Proving with Analytic Tableaux and Related Methods,* (1994).

[12] BUCHSBAUM, A. Um método automático de prova para a lógica paraconsistente. Dissertação de mestrado, Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro, 1988.

[13] BUCHSBAUM, A. *Lógica Geral*. 2004.

[14] CHANG, C. & LEE, R. C. *Symbolic Logic and Mechanical Theorem Proving*. Academic Press, New York, 1973.

[15] COPI, I. M. *Introdução à Lógica*. 2ª Edição, Editora Mestre Jou, São Paulo, 1978.

[16] DA COSTA, N. C. A. & KRAUSE, D. *Notas de Lógica*. Universidade Federal de Santa Catarina, Florianópolis - SC, 2004.

[17] DA COSTA, N. C. A. *Ensaio sobre os fundamentos da lógica*. Editora Hucitec: Ed. da Universidade de São Paulo, São Paulo, 1980.

[18] DE BRITO, P. F. Dedução automática por tableaux estruturada em XML. Dissertação de mestrado, Universidade Federal de Santa Catarina, Florianópolis - SC, 2003. Orientador: Arthur Buchsbaum.

[19] DE C. BACKX, A. & TAVARES, R. N. O. *Indução matemática, somatório, produtório*. ILTC-UFF, Niterói-RJ, 1984.

[20] DE OLIVEIRA CRUZ, A. J. Árvores. Disponível por WWW em URL:http://equipe.nce.ufrj.br/adriano/c/apostila/arvore.htm. Consultado em Julho 2003.

[21] ENDERTON, H. B. *A Mathematical Introduction to Logic*. Academic Press, Sant Diego - California, 1972.

[22] ENDERTON, H. B. *Elements of Set Theory*. Academic Press, 1977.

[23] FENDT, L. Refinamentos para o método dos tableaux. Universidade Federal de Santa Catarina - Dissertação de Mestrado, 2000. Orientador: Arthur Buchsbaum.

[24] FILHO, E. A. *Iniciação à Lógica Matemática*. Editora Nobel, São Paulo, 1986.

[25] GABBAY, E. D. M.; HOGGER, C. & ROBINSON, J. A. *Handbook of Logic in Artificial Intelligence and Logic Programming*. Clarendon Press - Oxford, New York - Toronto, 1993.

[26] GERSTING, J. L. *Fundamentos Matemáticos para a Ciência da Computação*. Editora LTC, Rio de Janeiro, 1982.

[27] HALMOS, P. R. *Teoria Ingênua dos Conjuntos*. Editora Ciência Moderna, São Paulo, 2001.

[28] HEGENBERG, L. *Lógica Simbólica*. Editora Herder - Universidade de São Paulo, São Paulo, 1966.

[29] HEGENBERG, L. *Lógica: o cálculo sentencial*. Editora Herder da Universidade de São Paulo, São Paulo, 1972.

[30] HEGENBERG, L. *Tabelas e Argumentos*. Editora Herder - Universidade de São Paulo, São Paulo, 1978.

[31] HINTIKKA, J. *Form and Content in Quantification Theory*. 8^a Edição, Acta Philosophica Fennica, Helsinki, 1955.

[32] JEFFREY, R. *Formal Logic - Its Scope and Limits*. McGraw-Hill, Princeton University, 1967.

[33] KNOBLAUCH, H. W. Quadro comparativo dos métodos dos tableaux e das resoluções. Universidade Federal de Santa Catarina - Dissertação de Mestrado, 2001. Orientador: Jorge Muniz Barreto.

[34] LEITSCH, A. *The Resolution Calculus*. Springer Verlag, 1997.

[35] LOVELAND, D. W. *Automated Theorem Proving: A Logical Basis*. North Holland, 1978.

[36] MANIN, Y. I. *A Course in Mathematical Logic*. Springer-Verlag, New York - Heidelberg, 1974.

[37] MONTEIRO, R. D. V. Uma proposta de aprendizagem interativa da lógica utilizando simuladores artificiais inteligentes. Dissertação de mestrado, Universidade Federal de Santa Catarina, Florianópolis - SC, 2002.

[38] MORTARI, C. A. *Introdução à Lógica*. UNESP: Imprensa Oficial do Estado, São Paulo, 2001.

[39] NERODE, A. & SHORE, R. A. *Logic for Applications*. 2^a Edição, Editora Springer, 1993.

[40] NOLT, J. & ROHATYN, D. *Lógica*. Schaum. McGraw-Hill, São Paulo, 1991.

[41] PINTO, J. S. *Tópicos de Matemática Discreta*. Universidade de Aveiro, 2003.

[42] ROISENBERG, M. Inteligência artificial. Disponível por WWW em URL:http://www.inf.ufsc.br/∼mauro/ine6102/6102.html. Consultado em Março de 2003.

[43] SEYMOUR, L. *Teoria dos Conjuntos*. Editora McGraw-HILL do Brasil, São Paulo, 1974.

[44] SMULLYAN, R. *First-Order Logic*. Dover Publications, New York, 1995.

[45] TENENBAUM, A. M. *Estruturas de Dados Usando C*. Makron Books, São Paulo, 1995.

[46] VELOSO, P.; DOS SANTOS, C.; AZEREDO, P. & FURTADO, A. *Estrutura de Dados*. Editora Campus, Rio de Janeiro, 1986.

[47] VILLAS, M. V.; DE MATTOS FERREIRA, A. G.; LEROY, P. G.; MIRANDA, C. & BOCKMAN, C. L. *Estrutura de dados: Conceitos e Técnicas de Implementação*. 5^a Edição, Editora Campus, Rio de Janeiro, 1993.

Índice Remissivo